On the Impact of Calibration Data in Post-training Quantization and Pruning

Miles Williams and Nikolaos Aletras
University of Sheffield
United Kingdom
{mwilliams15, n.aletras}@sheffield.ac.uk

Abstract

Quantization and pruning form the foundation of compression for neural networks, enabling efficient inference for large language models (LLMs). Recently, various quantization and pruning techniques have demonstrated remarkable performance in a post-training setting. They rely upon calibration data, a small set of unlabeled examples that are used to generate layer activations. However, no prior work has systematically investigated how the calibration data impacts the effectiveness of model compression methods. In this paper, we present the first extensive empirical study on the effect of calibration data upon LLM performance. We trial a variety of quantization and pruning methods, datasets, tasks, and models. Surprisingly, we find substantial variations in downstream task performance, contrasting existing work that suggests a greater level of robustness to the calibration data. Finally, we make a series of recommendations for the effective use of calibration data in LLM quantization and pruning.[1]

1 Introduction

Scaling is an essential component for unlocking new capabilities and improved performance in large language models (LLMs) (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023). However, the pursuit of scale has led to models that demand significant energy and computational resources (Strubell et al., 2019; Schwartz et al., 2020; Wu et al., 2022; Luccioni et al., 2023). Consequently, LLMs can be challenging to deploy, especially in resource-constrained environments (Dehghani et al., 2022; Menghani, 2023). These challenges have ultimately motivated a substantial body of research on model compression techniques, aiming to reduce computational demands while maintaining performance (Treviso et al., 2023).

[1] https://github.com/mlsw/llm-compression-calibration

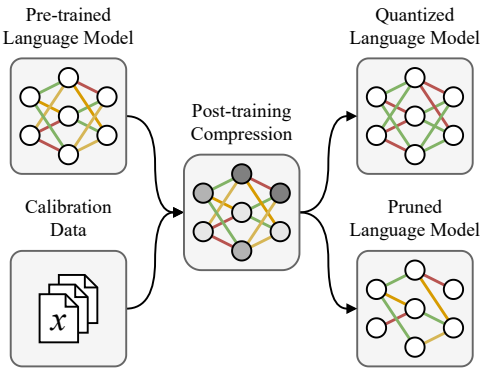


Figure 1: Post-training compression methods rely upon calibration data to generate layer activations.

Quantization and pruning are two of the most popular model compression techniques (Gholami et al., 2021; Hoefler et al., 2021). Pruning aims to remove redundant weights, while quantization seeks to represent weights (and possibly activations) in lower precision. Most recently, several quantization and pruning methods have demonstrated outstanding performance in a post-training setting (Frantar et al., 2023; Frantar and Alistarh, 2023; Dettmers et al., 2024; Sun et al., 2024).

Post-training compression techniques rely upon *calibration data* (Nagel et al., 2020; Hubara et al., 2021) to determine the distribution of layer activations. This process requires only a small number of examples, with further examples offering diminishing gains (Frantar and Alistarh, 2023; Sun et al., 2024). In the case of LLMs, the calibration set is routinely sampled from web text (Frantar et al., 2023; Frantar and Alistarh, 2023; Sun et al., 2024) or model pre-training data (Xiao et al., 2023; Dettmers et al., 2024). Notably, the calibration examples are sampled randomly. This is because post-training model compression methods are considered robust to the specific distribution of calibration data (Frantar and Alistarh, 2023; Sun et al., 2024; Dettmers et al., 2024).

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 10100–10118

In this paper, we present the first empirical study on the impact of calibration data used in post-training LLM compression. We offer an extensive study with several quantization and pruning methods, across a range of tasks, datasets, and models. Surprisingly, we find that downstream task performance can vary substantially according to the selected calibration data. This contrasts existing work, which suggests a high level of robustness. Finally, we offer a series of recommendations for the effective use of calibration data.

2 Related Work

2.1 Model Compression

Model compression is the process of reducing the memory requirements of a model, usually enabling improved inference efficiency (Treviso et al., 2023). In the case of neural networks, model compression has a rich history, with origins in seminal work from LeCun et al. (1989).[2] Quantization and pruning are two widely adopted approaches for model compression (Gholami et al., 2021; Hoefler et al., 2021). Pruning seeks to remove redundant weights, while quantization aims to represent the weights (and possibly activations) in lower precision. Applying these techniques to LLMs presents significant challenges, such as large-magnitude outlier features and high computational requirements (Dettmers et al., 2022; Frantar and Alistarh, 2023).

Post-training compression considers the scenario where a model must be compressed without retraining, instead relying upon a small amount of calibration data (Nagel et al., 2020; Hubara et al., 2021). While quantization and pruning are distinct methods, they are connected in a post-training setting via the *layer-wise compression problem* (Frantar and Alistarh, 2022). This involves selecting compressed weights for each layer that function closely to the original weights, with respect to the calibration data. More formally, given layer ℓ with weights $\mathbf{W}_\ell$ and inputs $\mathbf{X}_\ell$, the aim is to minimize $||\mathbf{W}_\ell\mathbf{X}_\ell - \widehat{\mathbf{W}}_\ell\mathbf{X}_\ell||_2^2$ with respect to the compressed weights $\widehat{\mathbf{W}}$. This is subject to a given compression constraint $\mathcal{C}(\widehat{\mathbf{W}}_\ell) > C$, which differs between quantization and pruning.

Recently, a range of approaches have been proposed for the layer-wise compression problem. SparseGPT (Frantar and Alistarh, 2023) enables LLM pruning up to 60% sparsity, with little impact upon perplexity. This sequentially prunes the weights in each column of the weight matrix using a series of Hessian inverses, followed by updating the remaining weights. Wanda (Sun et al., 2024) significantly improves upon the efficiency of SparseGPT, avoiding the expensive computation of the inverse Hessian. Instead, each layer is pruned according to the product of the weight magnitudes and ℓ_2 norm of the input activations. For quantization, GPTQ (Frantar et al., 2023) enables storing weights in 3- or 4-bit precision, similarly relying upon inverse Hessian information. SpQR (Dettmers et al., 2024) further enables practically lossless quantization of LLMs through identifying and holding outlier weights in higher precision. Other approaches include SmoothQuant (Xiao et al., 2023), which enables 8-bit quantization of weights and activations through shifting the complexity from the activations to the weights.

2.2 Sampling Calibration Data

Calibration data is customarily sampled from either web text (Frantar et al., 2023; Frantar and Alistarh, 2023; Sun et al., 2024) or model pre-training data (Xiao et al., 2023; Dettmers et al., 2024). The aforementioned approaches use little calibration data, commonly 128 examples, each comprising 2,048 tokens. The addition of further calibration samples provides diminishing gains (Frantar and Alistarh, 2023; Sun et al., 2024). Frantar and Alistarh (2023) demonstrate that SparseGPT reaches an optimal point with only 128 examples, while Sun et al. (2024) demonstrate that Wanda requires as few as eight examples.

Post-training compression methods are generally understood to be robust to the exact distribution of calibration examples (Frantar and Alistarh, 2023; Sun et al., 2024; Dettmers et al., 2024). However, concurrent to our own study, there has been a variety of recent work examining the impact of calibration data. Lee et al. (2023) note that question-answering performance can vary according to the calibration data sequence length when using GPTQ with 8-bit activation quantization. Moreover, Wu et al. (2023) find that GPTQ tends to overfit to the calibration data when assessing perplexity on several datasets. Separately, Jaiswal et al. (2024) suggest that curated calibration data could play an essential role in the design of improved LLM compression methods. Finally, Zeng et al. (2024) propose a novel calibration data sampling method to improve multilingual LLM compression.

[2]We refer interested readers to Frantar and Alistarh (2022) for a detailed treatment of quantization and pruning.

2.3 Evaluating Compressed Models

The recent demand for LLM compression methods has motivated a body of work investigating their broader effects upon model behavior and performance. Gonçalves and Strubell (2023) examine the issue of social bias in pre-trained language models, finding that post-training quantization has a regularizing effect. In a separate direction, Chrysostomou et al. (2023) investigate the impact of pruning upon hallucinations. Kuzmin et al. (2023) offer the first study that directly compares pruning against quantization. Excluding setups with extreme compression ratios, they find that quantization outperforms pruning. Perhaps the closest work to our own is from Jaiswal et al. (2024), which explores the failure modes of compressed LLMs. Notably, they observe that model compression has a substantial impact upon knowledge-intensive tasks.

3 Methodology

Our aim is to investigate the robustness of LLM pruning and quantization methods to the calibration data. To this end, we experiment with four compression methods and nine LLMs (i.e. three different sizes from three model families). We vary only the calibration data, trialing five source datasets, each comprising ten distinct calibration sets. This provides a total of 1,800 compressed models. We then evaluate each model across 11 standard NLP tasks, resulting in a total of 19,800 model evaluations.

3.1 Model Compression

As it is impractical to exhaustively test every model compression method, we select four popular approaches. Unless otherwise stated, we match the compression setup from the original work. We list complete hyperparameters in Appendix A, Table 3.

Quantization. For quantization, we consider **GPTQ** (Frantar et al., 2023) and **SpQR** (Dettmers et al., 2024). We follow Dettmers et al. (2024) in using 4-bit weight quantization, which offers an optimal balance between model size and performance (Dettmers and Zettlemoyer, 2023). Since GPTQ was proposed prior to the release of LLaMA, we use the hyperparameters from the AutoGPTQ library used by Hugging Face Transformers.[3]

Pruning. For pruning, we trial **SparseGPT** (Frantar and Alistarh, 2023) and **Wanda** (Sun et al., 2024). Since the goal of model compression is typically to enhance inference efficiency, we avoid unstructured pruning, which is challenging to accelerate (Wen et al., 2016). Instead, we opt for semi-structured sparsity, enabling significant inference speedups. We use the 2:4 sparsity pattern that is required for GPU acceleration, resulting in a sparsity ratio of 50% (Mishra et al., 2021).

[3] https://github.com/AutoGPTQ/AutoGPTQ

3.2 Evaluation Tasks

For a fair selection of evaluation tasks, we include all zero-shot tasks used in the original work of GPTQ, SpQR, SparseGPT, and Wanda. These are: (1) ARC easy (ARC-e) and (2) ARC challenge (ARC-c) sets (Clark et al., 2018); (3) BoolQ (Clark et al., 2019); (4) HellaSwag (Zellers et al., 2019); (5) LAMBADA (Paperno et al., 2016); (6) OpenBookQA (Banerjee et al., 2019); (7) PIQA (Bisk et al., 2020); (8) RTE (Dagan et al., 2006); (9) StoryCloze (Mostafazadeh et al., 2016); and (10) WinoGrande (Sakaguchi et al., 2021).

These zero-shot tasks primarily assess commonsense reasoning abilities, using binary, multiple choice, Cloze, and Winograd style questions. We report the evaluation set sizes in Appendix B.

Finally, following the evaluation protocol used by our selected model compression methods, we also report the model perplexity on the WikiText test set (Merity et al., 2017).

3.3 Calibration Data Sources

We explore a diverse variety of data sources to create our calibration sets. Following previous work (see Section 2), we include random web text and curated model pre-training datasets. To maintain the integrity of the zero-shot evaluations, we avoid using evaluation data as a source of calibration data, following Frantar et al. (2023). We therefore consider the following five data sources:

- **C4** (Raffel et al., 2020): We use the Colossal Clean Crawled Corpus as our baseline, following Frantar et al. (2023). This consists of web text from Common Crawl, filtered with multiple heuristics to form a subset of clean English text.

- **CNN-DM** (Hermann et al., 2015; See et al., 2017): The CNN/Daily Mail corpus consists of news articles from both publishers, covering a broad range of topics. We include this corpus since it provides a focused yet distinct genre of high-quality long-form text.

- **RedPajama** (Together Computer, 2023): Since the pre-training data for LLaMA is not publicly available, we instead use an open-source reproduction. This mainly consists of web text (Common Crawl and C4), in addition to selected high-quality sources such as arXiv, GitHub, Stack Exchange, and Books3 (Gao et al., 2020).
- **RefinedWeb** (Penedo et al., 2023): Assembled through stringent filtering and deduplication of Common Crawl, RefinedWeb is a curated model pre-training dataset. Penedo et al. (2023) find that models trained with this dataset exhibit superior zero-shot generalization abilities compared to alternatives such as The Pile (Gao et al., 2020).
- **Wikipedia**: We select English Wikipedia as a source of high-quality encyclopedic text. Specifically, we use a preprocessed and cleaned version of the dump from `2022-03-01`, prior to the "knowledge cutoff" of our selected models.

3.4 Models

We use three popular 'open-source' LLM families: (1) **LLaMA** (Touvron et al., 2023); (2) **Vicuna** (Chiang et al., 2023); and (3) **OPT** (Zhang et al., 2022). This includes base models (LLaMA and OPT) as well as instruction-tuned models (Vicuna).

Additionally, we select these three LLM families since they offer models of comparable sizes. For LLaMA and Vicuna, we select the 7B, 13B, and 33B model sizes. In the case of OPT, we select the closest comparable sizes (6.7B, 13B, and 30B).

3.5 Implementation Details

To create the calibration sets, we use the publicly available version of each source dataset from Hugging Face Datasets (Lhoest et al., 2021). Similarly, we use the weights and implementation of each model from Hugging Face Transformers (Wolf et al., 2020). To ensure that our model evaluations are robust and reproducible, we use the EleutherAI Language Model Evaluation Harness (Gao et al., 2021). Each model is compressed and evaluated using a single NVIDIA A100 (SXM 80GB) GPU.

3.6 Data Sampling

Previous work has demonstrated that increasing the number of calibration examples offers diminishing gains in language modeling performance (Frantar and Alistarh, 2023; Sun et al., 2024). We therefore follow existing work and randomly sample 128 calibration examples, each consisting of 2,048 tokens (Frantar et al., 2023; Frantar and Alistarh, 2023; Sun et al., 2024). This offers a total of 262,144 tokens in each calibration set. We provide a detailed analysis of how the quantity of calibration examples impacts performance in Section 4.

The use of random sampling avoids selection bias and ensures that each calibration set is representative of the source dataset. Similarly, we sample without replacement, to ensure that each calibration example appears only once. To examine the variability introduced by random sampling, we repeat the sampling process to create ten non-overlapping calibration sets for each source dataset. This provides a total of 50 distinct calibration sets.

Due to the vast size of C4, we follow Frantar et al. (2023) in sampling data from the first shard only. We use the same strategy for RefinedWeb, although for RedPajama we use the existing 1B token extract.[4]

4 Results & Analysis

Our principal aim is to investigate the impact of calibration data upon LLM performance. To this end, we examine the model performance across calibration sets, their source datasets, and evaluation tasks. Additionally, we explore trends in overall performance throughout the different configurations.

Performance can vary between calibration sets. Figure 2 shows the distribution of accuracy across ten calibration sets sampled from the same source dataset (C4) for each model in the LLaMA family. Across several tasks, we observe a substantial level of dispersion. In the most extreme cases with LLaMA-7B, we observe that the accuracy with SparseGPT ranges from 52.7% to 61.7% for RTE, and from 66.4% to 73.0% for BoolQ. We observe comparable levels of dispersion for the Vicuna and OPT model families, presented in Appendix C (Figures 5 and 6, respectively).

The degree of dispersion differs between tasks. We also observe that the dispersion of accuracy across calibration sets is elevated for certain tasks (Figure 2). For example, BoolQ and RTE present the highest levels of dispersion, with ranges up to 6.6% and 9.4%, respectively. However, we note that RTE has considerably fewer examples than the other tasks (Appendix B, Table 4). Consequently, RTE may offer a less stable estimate of dispersion.

[4] https://huggingface.co/datasets/togethercomputer/RedPajama-Data-1T-Sample

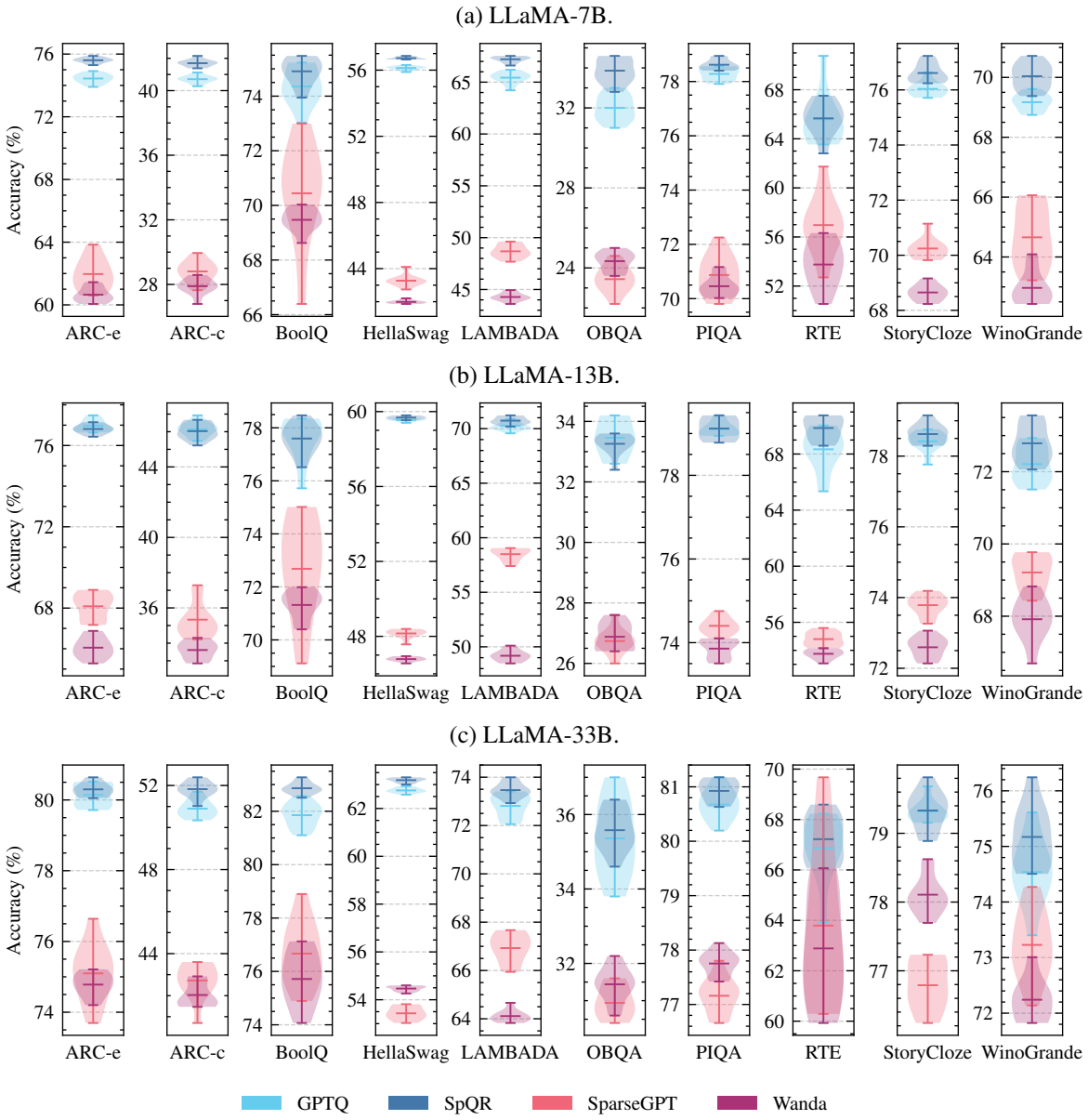


Figure 2: Distribution of accuracy across ten calibration sets sampled from C4 for the LLaMA family of models.

Certain data sources may outperform others. In the case of pruning, we observe that some calibration data sources achieve greater zero-shot performance for the same model. Table 1 shows the mean zero-shot accuracy and standard deviation across ten calibration sets sampled from each source dataset. For SparseGPT, we observe that RefinedWeb reaches the highest level of accuracy for eight of the nine models. In contrast, we observe that Wikipedia achieves the lowest performance across eight of the nine models. Finally, we note that Vicuna-7B demonstrates the greatest range between highest and lowest performing source dataset, with a mean zero-shot accuracy of 52.7% for CNN-DM versus 56.3% for RefinedWeb.

No data source consistently outperforms others. Although we observe that RefinedWeb generally performs well across various models, this is not universally the case. For example, LLaMA-7B pruned with Wanda achieves the highest mean zero-shot accuracy (52.7%) with RedPajama, yet the lowest (52.2%) with RefinedWeb (Table 1). Additionally, we emphasize that the difference between the source datasets with the highest and lowest mean zero-shot accuracy is often very small. Therefore, it is frequently unclear whether a source dataset outperforms its counterparts. This is especially relevant to quantization, where the difference is typically lowest. For example, this is usually around 0.2% for SpQR, across all model families and sizes.

Method	Dataset	LLaMA 7B	LLaMA 13B	LLaMA 33B	Vicuna 7B	Vicuna 13B	Vicuna 33B	OPT 6.7B	OPT 13B	OPT 30B
-	-	64.3	66.7	69.1	64.3	66.5	68.8	57.4	58.3	60.4
GPTQ	C4	$63.2_{0.2}$	$66.2_{0.2}$	$68.5_{0.2}$	$62.9_{0.2}$	$66.1_{0.2}$	$68.3_{0.2}$	$56.6_{0.4}$	$57.7_{0.3}$	$59.6_{0.2}$
	CNN-DM	$63.1_{0.3}$	$66.1_{0.2}$	$68.5_{0.2}$	$62.7_{0.3}$	$66.1_{0.2}$	$68.4_{0.2}$	$56.5_{0.3}$	$57.8_{0.3}$	$59.6_{0.1}$
	RedPajama	$63.2_{0.3}$	$66.2_{0.2}$	$68.6_{0.2}$	$63.2_{0.2}$	$66.2_{0.2}$	$68.4_{0.2}$	$56.7_{0.3}$	$57.9_{0.3}$	$59.8_{0.3}$
	RefinedWeb	$63.3_{0.3}$	$66.1_{0.2}$	$68.6_{0.2}$	$63.1_{0.2}$	$66.0_{0.2}$	$68.4_{0.2}$	$56.5_{0.4}$	$57.8_{0.2}$	$59.7_{0.2}$
	Wikipedia	$63.2_{0.2}$	$66.0_{0.2}$	$68.6_{0.2}$	$62.9_{0.3}$	$66.0_{0.1}$	$68.1_{0.2}$	$56.7_{0.3}$	$57.9_{0.2}$	$59.7_{0.2}$
SpQR	C4	$64.1_{0.2}$	$66.4_{0.1}$	$69.0_{0.2}$	$63.9_{0.1}$	$66.4_{0.2}$	$68.7_{0.1}$	$57.1_{0.1}$	$58.3_{0.1}$	$60.3_{0.2}$
	CNN-DM	$63.8_{0.2}$	$66.5_{0.2}$	$69.0_{0.1}$	$64.1_{0.1}$	$66.3_{0.1}$	$68.7_{0.2}$	$57.5_{0.2}$	$58.3_{0.3}$	$60.3_{0.1}$
	RedPajama	$64.0_{0.1}$	$66.4_{0.2}$	$68.9_{0.1}$	$64.1_{0.2}$	$66.5_{0.2}$	$68.7_{0.1}$	$57.2_{0.2}$	$58.3_{0.2}$	$60.3_{0.2}$
	RefinedWeb	$64.1_{0.2}$	$66.4_{0.2}$	$69.0_{0.1}$	$64.0_{0.2}$	$66.5_{0.2}$	$68.7_{0.2}$	$57.1_{0.1}$	$58.3_{0.1}$	$60.3_{0.2}$
	Wikipedia	$63.9_{0.2}$	$66.5_{0.2}$	$69.0_{0.1}$	$64.0_{0.2}$	$66.3_{0.2}$	$68.8_{0.1}$	$57.4_{0.1}$	$58.3_{0.2}$	$60.3_{0.2}$
SparseGPT	C4	$53.9_{0.4}$	$58.2_{0.3}$	$63.7_{0.5}$	$55.7_{0.4}$	$59.5_{0.4}$	$64.9_{0.3}$	$52.9_{0.3}$	$54.8_{0.3}$	$57.3_{0.3}$
	CNN-DM	$52.5_{0.4}$	$57.5_{0.2}$	$63.0_{0.4}$	$52.7_{0.6}$	$58.9_{0.3}$	$63.8_{0.3}$	$51.8_{0.3}$	$54.0_{0.4}$	$56.3_{0.2}$
	RedPajama	$53.3_{0.4}$	$57.6_{0.2}$	$63.2_{0.4}$	$55.5_{0.5}$	$59.3_{0.4}$	$64.5_{0.3}$	$52.6_{0.2}$	$54.4_{0.3}$	$56.9_{0.1}$
	RefinedWeb	$54.0_{0.4}$	$58.2_{0.2}$	$63.4_{0.6}$	$56.3_{0.4}$	$59.9_{0.3}$	$65.0_{0.5}$	$53.0_{0.2}$	$55.1_{0.1}$	$57.5_{0.1}$
	Wikipedia	$52.0_{0.5}$	$56.3_{0.3}$	$61.8_{0.4}$	$54.3_{0.6}$	$57.6_{0.6}$	$62.8_{0.3}$	$51.2_{0.2}$	$52.7_{0.2}$	$55.2_{0.2}$
Wanda	C4	$52.4_{0.3}$	$56.2_{0.2}$	$63.4_{0.3}$	$53.4_{0.4}$	$56.9_{0.2}$	$64.0_{0.2}$	$50.6_{0.2}$	$52.6_{0.2}$	$54.6_{0.3}$
	CNN-DM	$52.4_{0.2}$	$56.3_{0.1}$	$63.1_{0.1}$	$53.7_{0.3}$	$56.8_{0.2}$	$63.7_{0.1}$	$49.6_{0.2}$	$51.3_{0.4}$	$54.0_{0.3}$
	RedPajama	$52.7_{0.2}$	$56.3_{0.1}$	$63.1_{0.1}$	$53.3_{0.2}$	$57.0_{0.1}$	$63.8_{0.2}$	$50.5_{0.1}$	$52.4_{0.1}$	$54.8_{0.2}$
	RefinedWeb	$52.2_{0.3}$	$56.3_{0.2}$	$63.2_{0.1}$	$53.4_{0.3}$	$57.1_{0.1}$	$63.7_{0.2}$	$50.8_{0.1}$	$52.7_{0.1}$	$55.1_{0.3}$
	Wikipedia	$52.4_{0.2}$	$56.1_{0.2}$	$62.6_{0.2}$	$53.4_{0.3}$	$57.0_{0.1}$	$63.5_{0.2}$	$48.9_{0.2}$	$50.5_{0.2}$	$53.1_{0.2}$

Table 1: Mean accuracy of all zero-shot tasks across ten calibration sets. Standard deviation is denoted in subscript.

Calibration data may impact the overall model performance. To understand the extent to which calibration data can impact overall model performance, we examine the range between the best and worst performing calibration sets. Figure 4 presents the distribution of mean accuracy of all zero-shot tasks across all fifty calibration sets, for each model and compression method. We observe a non-negligible performance range across a variety of models and compression methods. For example, OPT-6.7B has a range of 1.6% for GPTQ, 0.9% for SpQR, 2.4% for SparseGPT, and 2.4% for Wanda.

Sensitivity can vary between models. We observe that the level of sensitivity can differ between models and their families (Figure 4). For pruning, SparseGPT exhibits a greater range (2.4-4.8%) across models than Wanda (0.6-2.9%). Interestingly, we also observe that Wanda exhibits a noticeably higher range for the OPT family (0.7-0.9%) compared to the LLaMA and Vicuna families (0.2-0.3%). In the case of quantization, we note that GPTQ displays a higher range (0.9-1.6%) than SpQR (0.6-1.0%). Again, we observe that the OPT family has a higher range for GPTQ (1.1-1.6%) than the LLaMA and Vicuna families (0.9-1.3%).

Quantization methods exhibit lower sensitivity. Considering both the individual task (Figures 2, 5, and 6) and overall performance (Figure 4), we observe that the quantization methods generally exhibit lower levels of dispersion than the pruning methods. Notably, the practically lossless compression offered by SpQR appears to leave little room for sensitivity to the calibration data. This suggests that while the exact distribution of calibration data may impact model performance, the compression method itself may be a more influential factor.

Pruning methods exhibit higher sensitivity. Contrasting the quantization methods, we observe that the pruning methods generally display higher dispersion. In particular, SparseGPT consistently demonstrates the highest levels of dispersion (Figure 4). We speculate that this is partly due to the destructive nature of pruning, in comparison to quantization. For example, Table 1 shows that the quantized models offer comparable performance to the original models, whereas there is a substantial performance degradation for the pruned models.

Additional calibration examples offer diminishing gains in language modeling performance. Figure 3 presents the perplexity on WikiText for LLaMA-7B compressed with increasing quantities of calibration data sampled from C4. For quantization, GPTQ improves from 6.13 ± 0.05 with a single example to 5.90 ± 0.03 with 128, while SpQR remains almost constant at 5.74 ± 0.01. In the case of pruning, SparseGPT gradually improves from 55.26 ± 10.82 with one example to 11.12 ± 0.26 with 128, while Wanda improves from 12.50 ± 0.15 to 11.56 ± 0.06. This corroborates findings from previous work that only a small number of examples are required to maximize language modeling performance (Frantar et al., 2023; Sun et al., 2024).

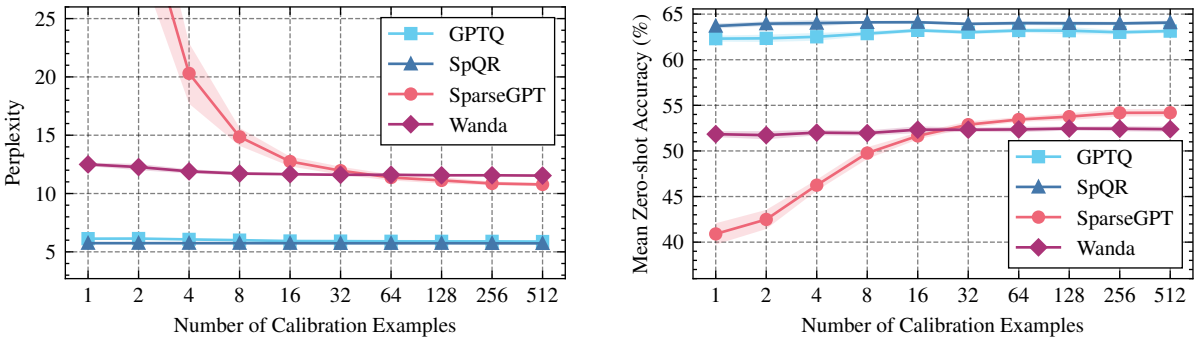


Figure 3: The perplexity on WikiText (L) and mean zero-shot accuracy (R) for LLaMA-7B with each compression method. We present the mean value and standard deviation (shaded) across ten calibration sets sampled from C4.

		LLaMA			Vicuna			OPT		
Method	Dataset	7B	13B	33B	7B	13B	33B	6.7B	13B	30B
-	-	5.68	5.09	4.10	6.90	6.09	5.22	10.86	10.13	9.56
GPTQ	C4	5.90$_{0.02}$	5.22$_{0.01}$	4.27$_{0.01}$	7.10$_{0.02}$	6.21$_{0.02}$	5.42$_{0.02}$	10.93$_{0.08}$	10.27$_{0.03}$	9.58$_{0.05}$
	CNN-DM	5.90$_{0.02}$	5.34$_{0.01}$	4.29$_{0.01}$	7.12$_{0.02}$	6.37$_{0.02}$	5.45$_{0.03}$	10.99$_{0.04}$	10.27$_{0.03}$	9.58$_{0.02}$
	RedPajama	5.91$_{0.02}$	5.22$_{0.01}$	4.27$_{0.01}$	7.10$_{0.04}$	6.22$_{0.02}$	5.43$_{0.02}$	10.99$_{0.05}$	10.27$_{0.03}$	9.55$_{0.07}$
	RefinedWeb	5.91$_{0.02}$	5.22$_{0.01}$	4.27$_{0.01}$	7.05$_{0.03}$	6.23$_{0.01}$	5.43$_{0.02}$	10.99$_{0.06}$	10.25$_{0.04}$	9.57$_{0.03}$
	Wikipedia	5.85$_{0.01}$	5.21$_{0.01}$	4.24$_{0.01}$	7.07$_{0.03}$	6.19$_{0.01}$	5.41$_{0.02}$	10.95$_{0.09}$	10.26$_{0.04}$	9.57$_{0.05}$
SpQR	C4	5.74$_{0.01}$	5.13$_{0.01}$	4.15$_{0.01}$	6.95$_{0.01}$	6.12$_{0.01}$	5.28$_{0.01}$	10.88$_{0.03}$	10.16$_{0.02}$	9.46$_{0.05}$
	CNN-DM	5.74$_{0.01}$	5.14$_{0.01}$	4.15$_{0.00}$	6.93$_{0.02}$	6.12$_{0.01}$	5.29$_{0.01}$	10.83$_{0.09}$	10.24$_{0.02}$	9.46$_{0.04}$
	RedPajama	5.73$_{0.00}$	5.13$_{0.01}$	4.15$_{0.00}$	6.93$_{0.02}$	6.12$_{0.01}$	5.28$_{0.01}$	10.90$_{0.02}$	10.16$_{0.02}$	9.45$_{0.04}$
	RefinedWeb	5.74$_{0.01}$	5.13$_{0.01}$	4.15$_{0.00}$	6.94$_{0.02}$	6.11$_{0.01}$	5.29$_{0.01}$	10.90$_{0.02}$	10.14$_{0.02}$	9.47$_{0.03}$
	Wikipedia	5.73$_{0.01}$	5.13$_{0.01}$	4.14$_{0.00}$	6.93$_{0.02}$	6.11$_{0.01}$	5.28$_{0.01}$	10.89$_{0.02}$	10.18$_{0.01}$	9.46$_{0.05}$
SparseGPT	C4	11.06$_{0.18}$	9.11$_{0.10}$	7.21$_{0.08}$	12.39$_{0.31}$	9.92$_{0.14}$	8.14$_{0.07}$	14.25$_{0.13}$	12.93$_{0.07}$	10.92$_{0.09}$
	CNN-DM	11.32$_{0.17}$	8.42$_{0.05}$	6.78$_{0.04}$	12.72$_{0.18}$	9.28$_{0.05}$	7.80$_{0.05}$	14.55$_{0.12}$	13.46$_{0.13}$	11.13$_{0.05}$
	RedPajama	10.71$_{0.17}$	8.82$_{0.09}$	7.03$_{0.07}$	12.04$_{0.13}$	9.49$_{0.10}$	8.06$_{0.07}$	14.15$_{0.11}$	13.02$_{0.11}$	11.18$_{0.06}$
	RefinedWeb	10.91$_{0.13}$	8.99$_{0.07}$	7.14$_{0.07}$	12.27$_{0.19}$	9.70$_{0.09}$	8.12$_{0.07}$	13.97$_{0.13}$	12.67$_{0.07}$	10.73$_{0.07}$
	Wikipedia	9.96$_{0.15}$	8.33$_{0.10}$	6.75$_{0.08}$	11.28$_{0.15}$	9.04$_{0.11}$	7.73$_{0.07}$	14.46$_{0.12}$	13.33$_{0.14}$	11.62$_{0.05}$
Wanda	C4	11.57$_{0.05}$	9.70$_{0.04}$	6.97$_{0.02}$	13.85$_{0.10}$	10.99$_{0.06}$	8.51$_{0.04}$	16.04$_{0.17}$	15.64$_{0.13}$	13.39$_{0.21}$
	CNN-DM	11.31$_{0.04}$	9.39$_{0.04}$	6.83$_{0.01}$	13.44$_{0.06}$	10.63$_{0.05}$	8.38$_{0.02}$	16.60$_{0.09}$	17.29$_{0.08}$	13.50$_{0.22}$
	RedPajama	11.38$_{0.05}$	9.45$_{0.05}$	6.90$_{0.02}$	13.51$_{0.14}$	10.62$_{0.04}$	8.45$_{0.02}$	16.22$_{0.11}$	15.99$_{0.18}$	13.63$_{0.20}$
	RefinedWeb	11.55$_{0.05}$	9.56$_{0.05}$	6.92$_{0.01}$	13.76$_{0.11}$	10.76$_{0.06}$	8.47$_{0.02}$	15.76$_{0.12}$	15.34$_{0.11}$	13.13$_{0.41}$
	Wikipedia	11.04$_{0.03}$	9.31$_{0.03}$	6.84$_{0.02}$	13.09$_{0.04}$	10.53$_{0.05}$	8.38$_{0.04}$	16.21$_{0.10}$	16.32$_{0.10}$	13.76$_{0.23}$

Table 2: Mean perplexity on WikiText across ten calibration sets. Standard deviation is denoted in subscript.

Additional calibration examples offer diminishing gains in zero-shot performance. Figure 3 also presents the mean zero-shot accuracy across tasks (Section 3.2) for LLaMA-7B compressed using an increasing number of calibration examples. For each compression method, we observe an identical trend to perplexity, with performance plateauing after only a small number of examples. Remarkably, GPTQ, SpQR, and Wanda demonstrate a consistent mean zero-shot accuracy with only a few examples. However, SparseGPT performance continues to improve beyond 128 examples, with quadruple the number of examples offering a 0.4% increase in mean zero-shot accuracy. This suggests that SparseGPT is best used with an expanded calibration set to maximize performance. However, it should be noted that increasing the number of calibration examples also increases the computational cost of compression (Frantar and Alistarh, 2023).

Perplexity can be challenging to interpret. Prior work relies on perplexity to assess robustness to the calibration data (Frantar and Alistarh, 2023; Sun et al., 2024). In practice, this consists of compressing a given model using a few different calibration sets, sampled from the same source dataset. Then, the difference in perplexity is measured using a standard dataset (often WikiText). Although this is an entirely reasonable approach, it can be challenging to interpret these results in the context of downstream task performance. Table 2, shows the WikiText test set perplexity and standard deviation across ten distinct calibration sets sampled from each source dataset. For example, applying SparseGPT to Vicuna-7B with the calibration sets from CNN-DM achieves a seemingly robust perplexity of 12.72±0.18. In contrast, the same models achieve 66.7%±4.7 on BoolQ, with accuracy ranging from 57.0% to 71.6%.

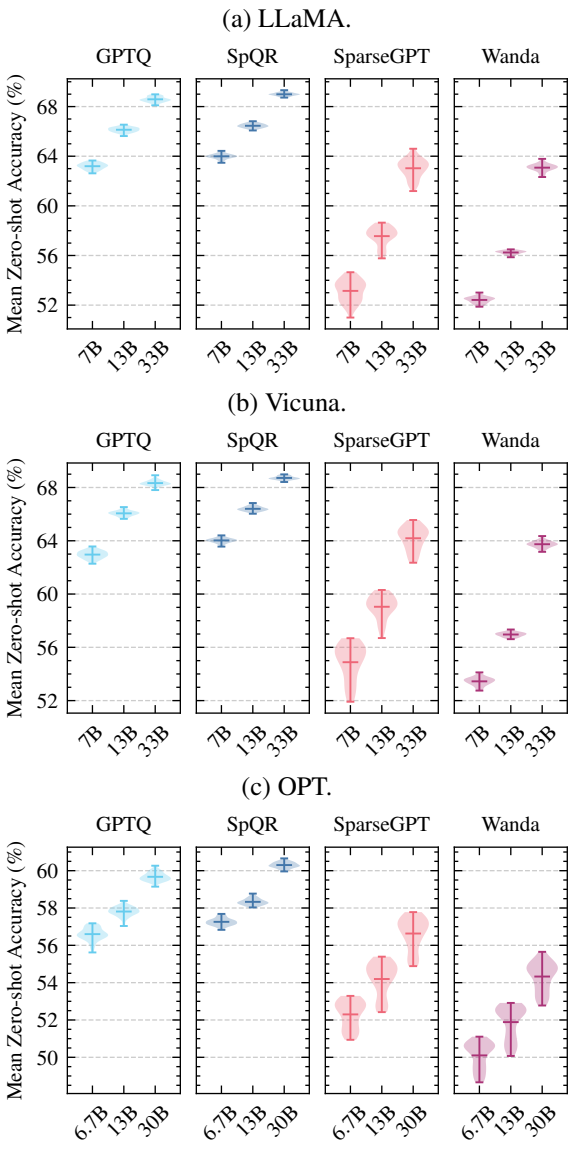


Figure 4: The distribution of mean zero-shot accuracy across all calibration sets for every configuration.

SparseGPT typically outperforms Wanda. Although the purpose of our study is not to compare model compression methods, we observe that SparseGPT mostly outperforms Wanda. Considering the C4 calibration data source that was used in both original works, we observe that SparseGPT achieves a higher mean zero-shot accuracy across all models. Moreover, we observe that the mean zero-shot accuracy achieved by SparseGPT is 2.2-2.6% higher across the OPT family, compared to 0.3-2.0% and 0.9-2.6% for LLaMA and Vicuna, respectively (Table 1). Although Sun et al. (2024) report mixed results for Wanda versus SparseGPT in the 2:4 semi-structured setting, we were somewhat surprised by the consistency and margin that SparseGPT outperforms Wanda in our experiments.

5 Recommendations

Our results suggest that calibration data used in post-training quantization and pruning can influence LLM performance. Consequently, we offer several recommendations concerning the use of calibration data to researchers and practitioners alike:

1. **Releasing calibration data**: Research relying upon calibration data for post-training model compression should release the data. As calibration data can ultimately affect model performance, this serves to improve reproducibility by removing a source of randomness.[5]

2. **Varying calibration data**: Evaluating downstream task performance across several calibration sets during model development can offer an insight into the sensitivity to the calibration data. This provides an opportunity to identify any issues with the compression setup or calibration data that may compromise model performance.

3. **Inspecting calibration data**: Randomly sampled calibration data can be manually inspected, to remove anomalous examples. Given the small number of examples comprising the calibration set, this may offer a practical way to maximize the performance of the compressed model.

6 Conclusion

In this paper, we presented the first extensive empirical study on calibration data for LLM quantization and pruning. We examined the downstream task performance across a variety of models, compression methods, and calibration data sources. Our results suggest that calibration data can substantially influence the performance of compressed LLMs. We supplement our findings with several recommendations for the effective use of calibration data.

We hope that our work will inspire further research surrounding the use of calibration data in LLM compression, an area which has seen limited attention. For future work, we are particularly interested in exploring how choices in the training protocol may influence the sensitivity to calibration data in LLM compression (Ahmadian et al., 2023).

[5]Many existing studies have released code to generate the calibration data, but not the calibration data itself. This is not robust since using a seed for random sampling does not guarantee consistency across software or platform versions.

Limitations

In this study, we rely entirely on English-language models, evaluation tasks, and calibration data. We operate under the assumption that the performance of the LLM compression methods we trial is generally language-agnostic. Nevertheless, we recognize the importance of linguistic diversity. We therefore hope to explore the performance of LLM compression methods across diverse language families (including low resource settings) in a future work.

Acknowledgments

We are grateful to George Chrysostomou, Huiyin Xue, and the anonymous reviewers for their invaluable feedback. MW is supported by the Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by UK Research and Innovation grant EP/S023062/1. NA is supported by EPSRC grant EP/Y009800/1, part of the RAI UK Keystone projects.

References

Arash Ahmadian, Saurabh Dash, Hongyu Chen, Bharat Venkitesh, Zhen Stephen Gou, Phil Blunsom, Ahmet Üstün, and Sara Hooker. 2023. Intriguing properties of quantization at scale. In *Advances in Neural Information Processing Systems*, volume 36, pages 34278–34294. Curran Associates, Inc.

Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019. Careful selection of knowledge to solve open book question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6120–6129, Florence, Italy. Association for Computational Linguistics.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. PaLM: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

George Chrysostomou, Zhixue Zhao, Miles Williams, and Nikolaos Aletras. 2023. Investigating hallucinations in pruned large language models for abstractive summarization. *Preprint*, arXiv:2311.09335.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *Preprint*, arXiv:1803.05457.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mostafa Dehghani, Yi Tay, Anurag Arnab, Lucas Beyer, and Ashish Vaswani. 2022. The efficiency misnomer. In *International Conference on Learning Representations*.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. GPT3.int8(): 8-bit matrix multiplication for transformers at scale. In *Advances in Neural Information Processing Systems*, volume 35, pages 30318–30332. Curran Associates, Inc.

Tim Dettmers, Ruslan A. Svirschevski, Vage Egiazarian, Denis Kuznedelev, Elias Frantar, Saleh Ashkboos, Alexander Borzunov, Torsten Hoefler, and Dan Alistarh. 2024. SpQR: A sparse-quantized representation for near-lossless LLM weight compression. In *The Twelfth International Conference on Learning Representations*.

Tim Dettmers and Luke Zettlemoyer. 2023. The case for 4-bit precision: k-bit inference scaling laws. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 7750–7774. PMLR.

Elias Frantar and Dan Alistarh. 2022. Optimal brain compression: A framework for accurate post-training quantization and pruning. In *Advances in Neural Information Processing Systems*, volume 35, pages 4475–4488. Curran Associates, Inc.

Elias Frantar and Dan Alistarh. 2023. SparseGPT: Massive language models can be accurately pruned in one-shot. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 10323–10337. PMLR.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2023. OPTQ: Accurate quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800GB dataset of diverse text for language modeling. *Preprint*, arXiv:2101.00027.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2021. A framework for few-shot language model evaluation.

Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. 2021. A survey of quantization methods for efficient neural network inference. *Preprint*, arXiv:2103.13630.

Gustavo Gonçalves and Emma Strubell. 2023. Understanding the effect of model compression on social bias in large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2663–2675, Singapore. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. 2021. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *Journal of Machine Learning Research*, 22(241):1–124.

Itay Hubara, Yury Nahshan, Yair Hanani, Ron Banner, and Daniel Soudry. 2021. Accurate post training quantization with small calibration sets. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4466–4475. PMLR.

Ajay Jaiswal, Zhe Gan, Xianzhi Du, Bowen Zhang, Zhangyang Wang, and Yinfei Yang. 2024. Compressing LLMs: The truth is rarely pure and never simple. In *The Twelfth International Conference on Learning Representations*.

Andrey Kuzmin, Markus Nagel, Mart van Baalen, Arash Behboodi, and Tijmen Blankevoort. 2023. Pruning vs quantization: Which is better? In *Advances in Neural Information Processing Systems*, volume 36, pages 62414–62427. Curran Associates, Inc.

Yann LeCun, John Denker, and Sara Solla. 1989. Optimal brain damage. In *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann.

Janghwan Lee, Minsoo Kim, Seungcheol Baek, Seok Hwang, Wonyong Sung, and Jungwook Choi. 2023. Enhancing computation efficiency in large language models through weight and activation quantization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14726–14739, Singapore. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. 2023. Estimating the carbon footprint

of BLOOM, a 176B parameter language model. *Journal of Machine Learning Research*, 24(253):1–15.

Gaurav Menghani. 2023. Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Comput. Surv.*, 55(12).

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *International Conference on Learning Representations*.

Asit Mishra, Jorge Albericio Latorre, Jeff Pool, Darko Stosic, Dusan Stosic, Ganesh Venkatesh, Chong Yu, and Paulius Micikevicius. 2021. Accelerating sparse deep neural networks. *Preprint*, arXiv:2104.08378.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. 2020. Up or down? adaptive rounding for post-training quantization. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Hamza Alobeidli, Alessandro Cappelli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data only. In *Advances in Neural Information Processing Systems*, volume 36, pages 79155–79172. Curran Associates, Inc.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. WinoGrande: An adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106.

Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. 2020. Green AI. *Commun. ACM*, 63(12):54–63.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. 2024. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations*.

Together Computer. 2023. RedPajama: An open source recipe to reproduce LLaMA training dataset.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Marcos Treviso, Ji-Ung Lee, Tianchu Ji, Betty van Aken, Qingqing Cao, Manuel R. Ciosici, Michael Hassid, Kenneth Heafield, Sara Hooker, Colin Raffel, Pedro H. Martins, André F. T. Martins, Jessica Zosa Forde, Peter Milder, Edwin Simpson, Noam Slonim, Jesse Dodge, Emma Strubell, Niranjan Balasubramanian, Leon Derczynski, Iryna Gurevych, and Roy Schwartz. 2023. Efficient methods for natural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 11:826–860.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. 2016. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, Michael Gschwind, Anurag Gupta, Myle Ott, Anastasia Melnikov, Salvatore Candido, David Brooks, Geeta Chauhan, Benjamin Lee, Hsien-Hsin Lee, Bugra Akyildiz, Maximilian Balandat, Joe Spisak, Ravi Jain, Mike Rabbat, and Kim Hazelwood. 2022. Sustainable AI: Environmental implications, challenges and opportunities. In *Proceedings of Machine Learning and Systems*, volume 4, pages 795–813.

Xiaoxia Wu, Haojun Xia, Stephen Youn, Zhen Zheng, Shiyang Chen, Arash Bakhtiari, Michael Wyatt, Reza Yazdani Aminabadi, Yuxiong He, Olatunji Ruwase, Leon Song, and Zhewei Yao. 2023. ZeroQuant(4+2): Redefining LLMs quantization with a new FP6-centric strategy for diverse generative tasks. *Preprint*, arXiv:2312.08583.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. 2023. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 38087–38099. PMLR.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Hongchuan Zeng, Hongshen Xu, Lu Chen, and Kai Yu. 2024. Multilingual brain surgeon: Large language models can be compressed leaving no language behind. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11794–11812, Torino, Italia. ELRA and ICCL.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open pre-trained transformer language models. *Preprint*, arXiv:2205.01068.

A Hyperparameters

Table 3 lists the complete hyperparameters used for each compression method. For SparseGPT, Wanda, and SpQR, these are taken from the original work. In the case of GPTQ, which was proposed prior to LLaMA, these are derived from AutoGPTQ.

B Datasets

Table 4 shows the number of examples for each evaluation task from the appropriate split (either test or validation).

Method	Hyperparameter	Value
GPTQ	Bits per Weight	4
	Dampening	0.01
	Descending Activation Order	Yes
	Group Size	128
	Symmetric Quantization	Yes
	True Sequential Quantization	Yes
SpQR	Bits per Scale	3
	Bits per Weight	4
	Bits per Zero	3
	Dampening	1.0
	Descending Activation Order	Yes
	Group Size (Weights)	16
	Group Size (Statistics)	16
	Symmetric Quantization	No
	True Sequential Quantization	No
	Outlier Threshold	0.2
SparseGPT	Dampening	0.01
	Group Size	128
	Sparsity	2:4
Wanda	Group Size	1
	Sparsity	2:4

Table 3: The hyperparameters used for all experiments.

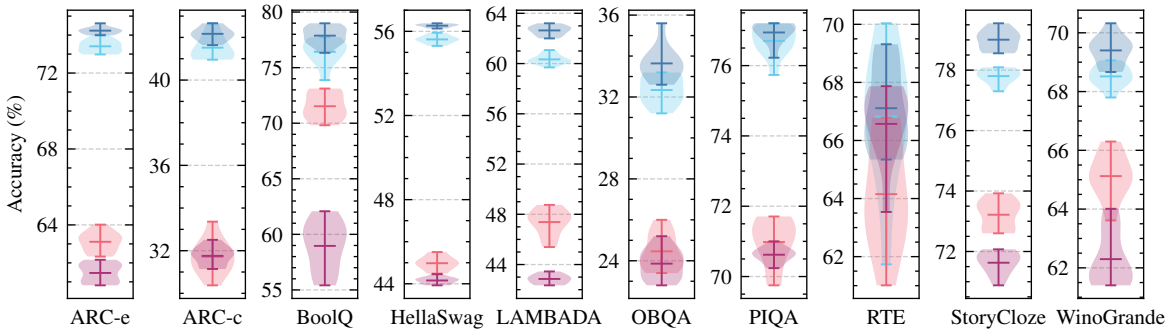
Dataset	# Examples
ARC-Easy (Clark et al., 2018)	2,376
ARC-Challenge (Clark et al., 2018)	1,172
BoolQ (Clark et al., 2019)	3,270
HellaSwag (Zellers et al., 2019)	10,042
LAMBADA (Paperno et al., 2016)	5,153
OpenBookQA (Banerjee et al., 2019)	500
PIQA (Bisk et al., 2020)	1,838
RTE (Dagan et al., 2006)	277
StoryCloze (Mostafazadeh et al., 2016)	1,511
WinoGrande (Sakaguchi et al., 2021)	1,267

Table 4: Number of examples for each evaluation task.

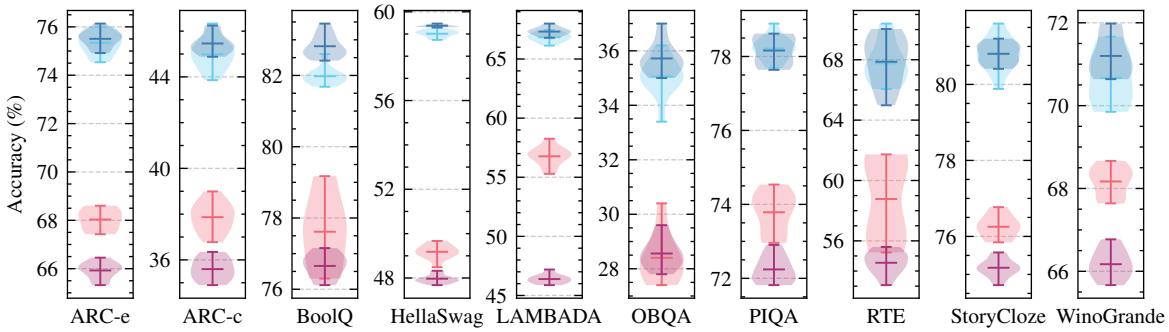
C Complete Results

In addition to the summarized results (Section 4), we provide a complete tabulation of results for all models, tasks, compression methods, and source datasets. We present the mean accuracy and standard deviation across ten distinct calibration sets for each model family: LLaMA 7B, 13B, and 33B (Tables 5, 6 and 7); Vicuna 7B, 13B, and 33B (Tables 8, 9, and 10); and OPT 6.7B, 13B, and 30B (Tables 11, 12, and 13). Finally, we present the distribution of accuracy across calibration sets sampled from C4 (Figure 2) for the Vicuna and OPT model families in Figures 6 and 5, respectively.

(a) Vicuna-7B.



(b) Vicuna-13B.



(c) Vicuna-33B.

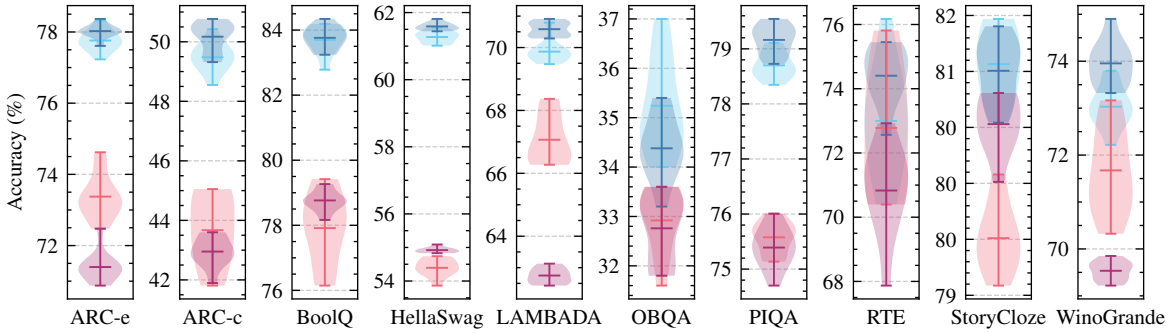
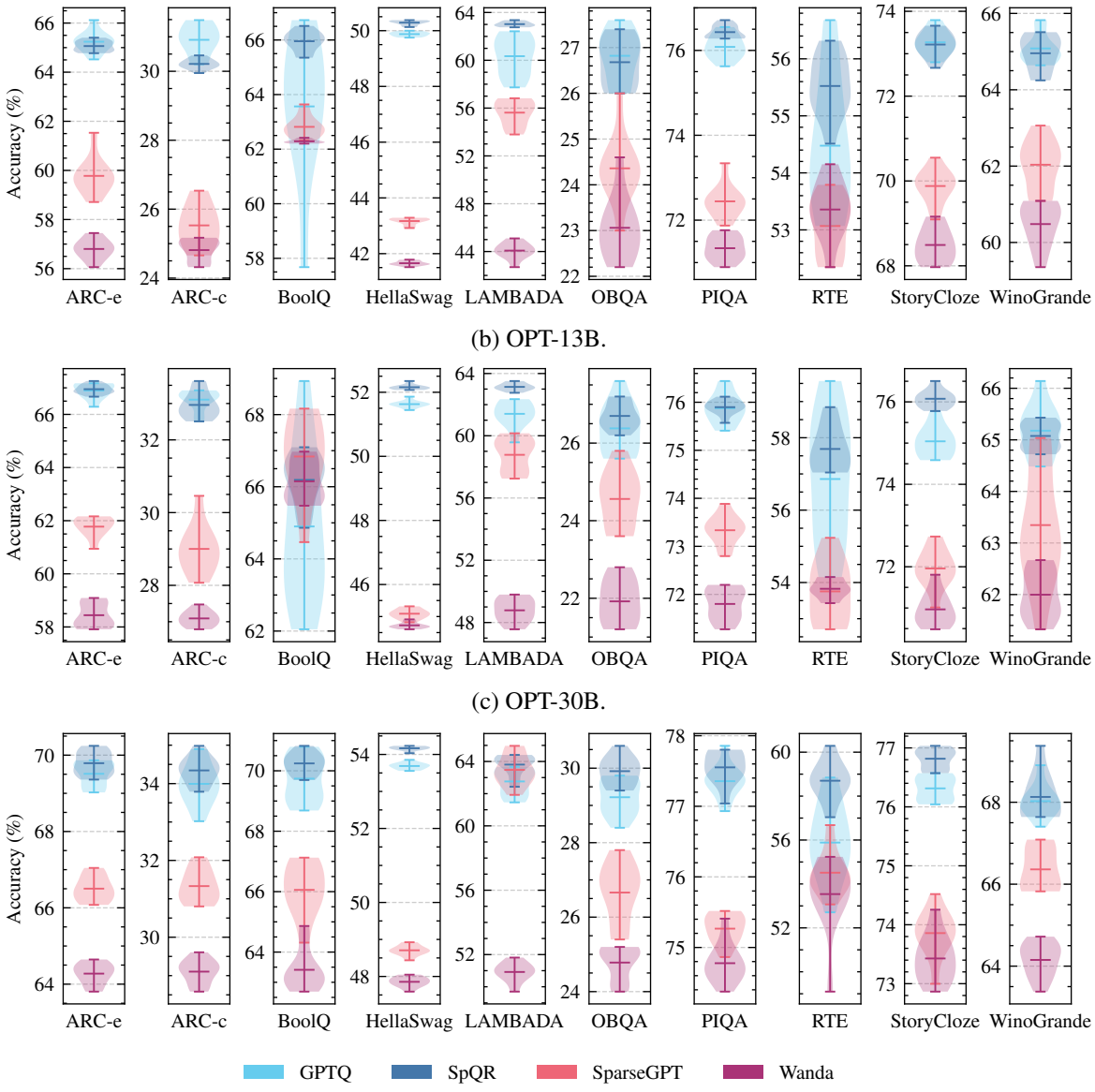


Figure 5: Distribution of accuracy across ten calibration sets sampled from C4 for the Vicuna family of models.

(a) OPT-6.7B.



(b) OPT-13B.

(c) OPT-30B.

Figure 6: Distribution of accuracy across ten calibration sets sampled from C4 for the OPT family of models.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	75.3	41.9	75.1	57.0	67.8	34.4	78.7	66.4	76.7	69.9	64.3
GPTQ	C4	74.4$_{0.3}$	40.7$_{0.2}$	74.4$_{0.7}$	56.1$_{0.1}$	65.4$_{0.6}$	32.0$_{0.7}$	78.3$_{0.2}$	65.6$_{2.1}$	76.0$_{0.3}$	69.2$_{0.3}$	63.2$_{0.2}$
	CNN-DM	74.3$_{0.3}$	40.9$_{0.5}$	74.1$_{0.5}$	56.1$_{0.1}$	64.7$_{0.6}$	32.1$_{0.9}$	78.3$_{0.3}$	65.7$_{1.3}$	75.8$_{0.2}$	69.2$_{0.8}$	63.1$_{0.3}$
	RedPajama	74.1$_{0.4}$	40.5$_{0.7}$	74.0$_{0.5}$	56.2$_{0.1}$	65.3$_{0.6}$	31.7$_{0.8}$	78.4$_{0.2}$	66.5$_{2.6}$	76.0$_{0.4}$	68.8$_{0.5}$	63.2$_{0.3}$
	RefinedWeb	74.0$_{0.5}$	40.9$_{0.6}$	74.6$_{0.6}$	56.3$_{0.1}$	65.6$_{0.4}$	32.0$_{1.0}$	78.4$_{0.3}$	65.4$_{2.2}$	76.0$_{0.2}$	69.5$_{0.4}$	63.3$_{0.3}$
	Wikipedia	74.4$_{0.3}$	41.4$_{0.9}$	74.4$_{0.5}$	56.3$_{0.2}$	65.0$_{0.4}$	31.7$_{0.8}$	78.5$_{0.2}$	64.6$_{2.3}$	76.2$_{0.4}$	69.5$_{0.4}$	63.2$_{0.2}$
SpQR	C4	75.6$_{0.2}$	41.7$_{0.2}$	74.9$_{0.5}$	56.7$_{0.1}$	67.2$_{0.3}$	33.9$_{0.6}$	78.6$_{0.2}$	65.7$_{1.4}$	76.6$_{0.3}$	70.0$_{0.4}$	64.1$_{0.2}$
	CNN-DM	75.2$_{0.2}$	41.7$_{0.4}$	74.5$_{0.5}$	56.7$_{0.1}$	66.9$_{0.1}$	33.4$_{0.6}$	78.5$_{0.2}$	65.4$_{1.0}$	76.3$_{0.2}$	69.7$_{0.5}$	63.8$_{0.2}$
	RedPajama	75.3$_{0.3}$	41.8$_{0.4}$	74.8$_{0.5}$	56.7$_{0.1}$	67.1$_{0.4}$	33.0$_{0.4}$	78.6$_{0.2}$	66.1$_{1.0}$	76.5$_{0.2}$	70.2$_{0.4}$	64.0$_{0.1}$
	RefinedWeb	75.3$_{0.4}$	41.5$_{0.3}$	74.7$_{0.6}$	56.7$_{0.1}$	67.3$_{0.3}$	33.9$_{0.5}$	78.6$_{0.3}$	66.1$_{1.4}$	76.7$_{0.3}$	70.1$_{0.4}$	64.1$_{0.2}$
	Wikipedia	75.3$_{0.3}$	41.5$_{0.4}$	74.7$_{0.8}$	56.7$_{0.1}$	67.2$_{0.3}$	33.0$_{0.7}$	78.4$_{0.3}$	66.0$_{1.3}$	76.8$_{0.2}$	69.9$_{0.4}$	63.9$_{0.2}$
SparseGPT	C4	62.0$_{1.0}$	28.8$_{0.7}$	70.4$_{1.9}$	43.3$_{0.3}$	48.7$_{0.7}$	23.4$_{0.7}$	70.9$_{0.8}$	57.0$_{2.4}$	70.2$_{0.4}$	64.7$_{1.0}$	53.9$_{0.4}$
	CNN-DM	58.3$_{1.1}$	27.0$_{0.4}$	69.5$_{1.3}$	42.2$_{0.2}$	47.9$_{1.1}$	22.1$_{1.0}$	69.0$_{0.4}$	54.4$_{3.2}$	69.8$_{0.5}$	64.7$_{0.9}$	52.5$_{0.4}$
	RedPajama	61.4$_{0.8}$	28.4$_{0.7}$	70.1$_{2.6}$	42.2$_{0.4}$	48.5$_{1.6}$	22.9$_{0.9}$	69.5$_{0.4}$	57.0$_{3.1}$	69.3$_{0.4}$	64.2$_{0.9}$	53.3$_{0.4}$
	RefinedWeb	60.9$_{0.8}$	28.1$_{0.8}$	70.5$_{1.9}$	43.0$_{0.5}$	51.3$_{1.2}$	22.5$_{1.1}$	70.8$_{0.6}$	57.2$_{2.1}$	70.1$_{0.7}$	65.3$_{0.8}$	54.0$_{0.4}$
	Wikipedia	60.7$_{1.1}$	28.6$_{0.7}$	65.7$_{1.7}$	41.2$_{0.4}$	45.5$_{1.7}$	23.2$_{0.7}$	69.3$_{0.5}$	54.4$_{3.2}$	67.9$_{0.4}$	63.6$_{0.6}$	52.0$_{0.5}$
Wanda	C4	60.6$_{0.5}$	27.9$_{0.5}$	69.5$_{0.4}$	42.0$_{0.1}$	44.3$_{0.4}$	24.3$_{0.5}$	70.5$_{0.4}$	53.8$_{1.9}$	68.6$_{0.3}$	63.0$_{0.6}$	52.4$_{0.3}$
	CNN-DM	59.5$_{0.6}$	27.0$_{0.3}$	69.3$_{0.4}$	41.9$_{0.1}$	44.1$_{0.3}$	23.8$_{0.6}$	69.8$_{0.3}$	57.3$_{1.5}$	67.9$_{0.3}$	63.1$_{0.5}$	52.4$_{0.2}$
	RedPajama	61.2$_{0.5}$	28.0$_{0.4}$	69.3$_{0.4}$	41.7$_{0.1}$	44.5$_{0.3}$	24.3$_{0.3}$	70.1$_{0.2}$	57.0$_{1.6}$	68.4$_{0.2}$	62.2$_{0.5}$	52.7$_{0.2}$
	RefinedWeb	60.6$_{0.5}$	27.8$_{0.4}$	69.1$_{0.2}$	41.9$_{0.2}$	45.0$_{0.3}$	24.0$_{0.5}$	70.4$_{0.5}$	52.6$_{2.1}$	68.4$_{0.2}$	62.2$_{0.5}$	52.2$_{0.3}$
	Wikipedia	60.0$_{0.5}$	27.8$_{0.6}$	69.0$_{0.4}$	41.6$_{0.2}$	43.4$_{0.3}$	24.0$_{0.6}$	69.8$_{0.2}$	58.2$_{1.2}$	68.0$_{0.4}$	62.4$_{0.4}$	52.4$_{0.2}$

Table 5: Mean accuracy across ten calibration sets for LLaMA-7B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	77.4	46.5	78.0	59.9	71.1	33.2	79.2	70.4	78.5	72.8	66.7
GPTQ	C4	77.0$_{0.3}$	46.1$_{0.4}$	77.6$_{0.8}$	59.6$_{0.1}$	70.3$_{0.4}$	33.5$_{0.6}$	79.1$_{0.2}$	68.3$_{1.5}$	78.4$_{0.3}$	72.2$_{0.5}$	66.2$_{0.2}$
	CNN-DM	76.9$_{0.4}$	46.0$_{0.7}$	77.8$_{0.4}$	59.4$_{0.2}$	70.4$_{0.3}$	33.1$_{0.6}$	78.9$_{0.2}$	68.0$_{2.0}$	78.4$_{0.3}$	72.4$_{0.4}$	66.1$_{0.2}$
	RedPajama	77.2$_{0.4}$	46.2$_{0.7}$	77.4$_{0.6}$	59.6$_{0.2}$	70.3$_{0.2}$	33.5$_{0.6}$	79.1$_{0.2}$	68.0$_{1.9}$	78.4$_{0.2}$	72.4$_{0.5}$	66.2$_{0.2}$
	RefinedWeb	77.0$_{0.4}$	46.2$_{0.4}$	77.6$_{0.6}$	59.5$_{0.1}$	70.4$_{0.4}$	33.6$_{0.3}$	79.2$_{0.3}$	67.3$_{1.5}$	78.3$_{0.3}$	72.4$_{0.4}$	66.1$_{0.2}$
	Wikipedia	76.9$_{0.4}$	46.1$_{0.4}$	77.0$_{0.7}$	59.4$_{0.2}$	70.4$_{0.6}$	33.4$_{0.5}$	79.0$_{0.3}$	67.4$_{1.8}$	78.1$_{0.3}$	72.2$_{0.5}$	66.0$_{0.2}$
SpQR	C4	76.8$_{0.2}$	46.0$_{0.5}$	77.6$_{0.6}$	59.7$_{0.1}$	70.7$_{0.3}$	33.3$_{0.4}$	79.1$_{0.2}$	69.9$_{0.8}$	78.6$_{0.3}$	72.8$_{0.5}$	66.4$_{0.1}$
	CNN-DM	77.0$_{0.2}$	46.1$_{0.4}$	78.0$_{0.4}$	59.7$_{0.1}$	70.9$_{0.3}$	33.2$_{0.5}$	79.0$_{0.2}$	69.6$_{1.7}$	78.4$_{0.3}$	72.6$_{0.4}$	66.5$_{0.2}$
	RedPajama	76.8$_{0.2}$	46.3$_{0.3}$	77.8$_{0.2}$	59.6$_{0.1}$	70.8$_{0.2}$	33.3$_{0.4}$	79.1$_{0.2}$	69.3$_{1.6}$	78.3$_{0.2}$	72.9$_{0.5}$	66.4$_{0.2}$
	RefinedWeb	77.0$_{0.3}$	46.3$_{0.4}$	77.9$_{0.4}$	59.7$_{0.1}$	70.9$_{0.3}$	33.6$_{0.4}$	79.2$_{0.2}$	68.7$_{1.6}$	78.4$_{0.3}$	72.8$_{0.4}$	66.4$_{0.2}$
	Wikipedia	77.0$_{0.2}$	46.2$_{0.5}$	78.1$_{0.4}$	59.7$_{0.1}$	70.9$_{0.4}$	33.7$_{0.5}$	79.0$_{0.1}$	69.4$_{1.7}$	78.5$_{0.2}$	72.8$_{0.4}$	66.5$_{0.2}$
SparseGPT	C4	68.1$_{0.7}$	35.3$_{1.0}$	72.7$_{1.9}$	48.2$_{0.2}$	58.5$_{0.6}$	26.7$_{0.4}$	74.4$_{0.2}$	54.8$_{0.5}$	73.8$_{0.3}$	69.2$_{0.5}$	58.2$_{0.3}$
	CNN-DM	65.5$_{0.6}$	34.4$_{0.7}$	72.8$_{1.5}$	47.7$_{0.2}$	58.9$_{0.5}$	26.8$_{0.7}$	73.4$_{0.7}$	53.5$_{0.5}$	73.5$_{0.4}$	68.5$_{0.7}$	57.5$_{0.2}$
	RedPajama	67.6$_{0.8}$	34.7$_{0.8}$	72.7$_{1.4}$	47.1$_{0.4}$	58.5$_{0.6}$	26.2$_{0.8}$	73.1$_{0.5}$	54.5$_{1.1}$	73.2$_{0.3}$	68.6$_{0.7}$	57.6$_{0.2}$
	RefinedWeb	67.4$_{0.6}$	34.7$_{0.9}$	73.1$_{0.9}$	48.0$_{0.3}$	61.6$_{0.8}$	26.5$_{0.8}$	74.0$_{0.6}$	54.7$_{1.0}$	73.7$_{0.3}$	68.7$_{0.6}$	58.2$_{0.2}$
	Wikipedia	66.6$_{0.9}$	33.3$_{0.9}$	69.9$_{1.7}$	46.4$_{0.2}$	53.9$_{0.7}$	26.8$_{0.9}$	72.5$_{0.6}$	53.1$_{0.4}$	72.2$_{0.3}$	67.8$_{0.9}$	56.3$_{0.3}$
Wanda	C4	66.1$_{0.5}$	33.6$_{0.5}$	71.3$_{0.5}$	46.8$_{0.1}$	49.2$_{0.5}$	26.9$_{0.5}$	73.9$_{0.2}$	53.8$_{0.4}$	72.6$_{0.3}$	67.9$_{0.6}$	56.2$_{0.2}$
	CNN-DM	66.2$_{0.4}$	33.5$_{0.5}$	71.9$_{0.4}$	46.8$_{0.1}$	50.9$_{0.3}$	27.0$_{0.6}$	73.1$_{0.3}$	53.5$_{0.3}$	72.8$_{0.2}$	67.8$_{0.5}$	56.3$_{0.1}$
	RedPajama	66.2$_{0.4}$	33.9$_{0.5}$	71.6$_{0.5}$	46.5$_{0.1}$	49.6$_{0.3}$	27.1$_{0.5}$	73.2$_{0.3}$	54.4$_{0.4}$	72.4$_{0.2}$	68.0$_{0.4}$	56.3$_{0.1}$
	RefinedWeb	65.9$_{0.4}$	33.5$_{0.5}$	71.9$_{0.4}$	46.7$_{0.2}$	49.4$_{0.4}$	27.0$_{0.6}$	73.6$_{0.3}$	54.2$_{0.2}$	72.5$_{0.3}$	68.1$_{0.6}$	56.3$_{0.2}$
	Wikipedia	66.8$_{0.4}$	34.1$_{0.2}$	71.2$_{1.0}$	46.2$_{0.1}$	48.9$_{0.5}$	26.4$_{0.5}$	73.1$_{0.2}$	53.9$_{0.5}$	72.4$_{0.2}$	67.8$_{0.5}$	56.1$_{0.2}$

Table 6: Mean accuracy across ten calibration sets for LLaMA-13B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	80.4	52.8	82.6	63.3	73.4	36.0	81.0	66.8	79.1	75.9	69.1
GPTQ	C4	80.1$_{0.3}$	50.9$_{0.4}$	81.9$_{0.5}$	62.8$_{0.1}$	72.8$_{0.5}$	35.4$_{1.0}$	80.7$_{0.3}$	66.9$_{1.3}$	79.3$_{0.2}$	74.5$_{0.6}$	68.5$_{0.2}$
	CNN-DM	79.7$_{0.5}$	50.4$_{0.4}$	82.2$_{0.7}$	62.6$_{0.1}$	73.0$_{0.4}$	35.2$_{0.7}$	80.7$_{0.2}$	67.2$_{1.9}$	79.2$_{0.3}$	75.2$_{0.4}$	68.5$_{0.2}$
	RedPajama	79.9$_{0.4}$	50.7$_{0.8}$	81.7$_{0.7}$	62.7$_{0.1}$	73.1$_{0.2}$	35.2$_{0.5}$	80.6$_{0.2}$	68.0$_{1.5}$	79.1$_{0.2}$	74.9$_{0.4}$	68.6$_{0.2}$
	RefinedWeb	79.5$_{0.3}$	50.7$_{0.4}$	82.3$_{0.4}$	62.7$_{0.1}$	72.9$_{0.5}$	35.2$_{0.7}$	80.8$_{0.3}$	68.1$_{1.5}$	79.3$_{0.2}$	74.9$_{0.3}$	68.6$_{0.2}$
	Wikipedia	80.1$_{0.4}$	50.6$_{0.7}$	82.0$_{0.7}$	62.7$_{0.1}$	73.0$_{0.4}$	35.0$_{0.6}$	80.7$_{0.3}$	67.4$_{1.7}$	79.4$_{0.4}$	75.4$_{0.4}$	68.6$_{0.2}$
SpQR	C4	80.3$_{0.2}$	51.8$_{0.4}$	82.9$_{0.2}$	63.2$_{0.1}$	73.5$_{0.3}$	35.6$_{0.6}$	80.9$_{0.2}$	67.2$_{0.9}$	79.3$_{0.3}$	75.2$_{0.6}$	69.0$_{0.2}$
	CNN-DM	80.4$_{0.3}$	51.7$_{0.5}$	82.7$_{0.3}$	63.1$_{0.1}$	73.4$_{0.2}$	35.8$_{0.6}$	81.0$_{0.2}$	67.7$_{0.5}$	79.2$_{0.1}$	75.3$_{0.5}$	69.0$_{0.1}$
	RedPajama	80.4$_{0.1}$	51.9$_{0.5}$	82.6$_{0.3}$	63.1$_{0.1}$	73.4$_{0.2}$	35.6$_{0.5}$	80.9$_{0.2}$	66.8$_{0.8}$	79.2$_{0.1}$	75.3$_{0.3}$	68.9$_{0.1}$
	RefinedWeb	80.6$_{0.2}$	51.8$_{0.4}$	82.7$_{0.3}$	63.1$_{0.1}$	73.6$_{0.2}$	35.7$_{0.5}$	80.9$_{0.2}$	67.3$_{0.4}$	79.2$_{0.1}$	75.1$_{0.2}$	69.0$_{0.1}$
	Wikipedia	80.5$_{0.3}$	51.7$_{0.3}$	82.6$_{0.3}$	63.2$_{0.1}$	73.4$_{0.3}$	35.9$_{0.3}$	80.9$_{0.2}$	67.4$_{1.0}$	79.2$_{0.2}$	75.2$_{0.3}$	69.0$_{0.1}$
SparseGPT	C4	75.1$_{0.8}$	42.7$_{0.9}$	76.7$_{1.2}$	53.4$_{0.3}$	66.9$_{0.6}$	30.9$_{0.4}$	77.2$_{0.3}$	63.8$_{3.2}$	76.8$_{0.4}$	73.2$_{0.6}$	63.7$_{0.5}$
	CNN-DM	73.4$_{0.3}$	40.8$_{0.6}$	76.7$_{1.5}$	53.1$_{0.2}$	67.3$_{0.5}$	30.2$_{1.1}$	76.5$_{0.4}$	63.5$_{2.0}$	76.5$_{0.5}$	72.1$_{0.9}$	63.0$_{0.4}$
	RedPajama	74.5$_{0.5}$	42.0$_{0.7}$	76.9$_{1.8}$	52.4$_{0.2}$	67.9$_{0.5}$	30.4$_{1.1}$	76.4$_{0.5}$	62.9$_{3.4}$	75.9$_{0.4}$	72.8$_{0.6}$	63.2$_{0.4}$
	RefinedWeb	74.6$_{0.6}$	42.2$_{1.0}$	75.9$_{2.0}$	53.1$_{0.3}$	68.9$_{0.7}$	30.9$_{0.9}$	77.2$_{0.3}$	61.8$_{4.0}$	76.5$_{0.5}$	73.1$_{0.7}$	63.4$_{0.6}$
	Wikipedia	74.0$_{0.6}$	40.8$_{0.7}$	71.2$_{1.2}$	51.7$_{0.2}$	65.5$_{0.6}$	30.1$_{0.9}$	75.7$_{0.2}$	63.1$_{2.2}$	74.9$_{0.4}$	71.5$_{0.8}$	61.8$_{0.4}$
Wanda	C4	74.8$_{0.3}$	42.0$_{0.6}$	75.7$_{1.0}$	54.5$_{0.1}$	64.1$_{0.2}$	31.4$_{0.5}$	77.8$_{0.2}$	62.9$_{2.2}$	78.1$_{0.3}$	72.2$_{0.4}$	63.4$_{0.3}$
	CNN-DM	73.5$_{0.3}$	41.2$_{0.6}$	75.5$_{0.6}$	54.2$_{0.1}$	64.4$_{0.2}$	31.3$_{0.6}$	77.2$_{0.2}$	62.8$_{2.2}$	77.8$_{0.2}$	72.7$_{0.5}$	63.1$_{0.1}$
	RedPajama	74.7$_{0.4}$	42.0$_{0.6}$	74.8$_{0.5}$	54.0$_{0.1}$	64.4$_{0.4}$	30.4$_{0.5}$	77.4$_{0.2}$	64.3$_{1.7}$	77.6$_{0.3}$	71.7$_{0.6}$	63.1$_{0.1}$
	RefinedWeb	74.5$_{0.3}$	41.6$_{0.4}$	75.3$_{0.6}$	54.3$_{0.1}$	64.7$_{0.3}$	31.0$_{0.8}$	77.7$_{0.2}$	63.2$_{1.4}$	77.9$_{0.3}$	72.2$_{0.3}$	63.2$_{0.1}$
	Wikipedia	73.8$_{0.3}$	41.3$_{0.3}$	72.2$_{0.9}$	53.7$_{0.1}$	64.1$_{0.3}$	30.3$_{0.5}$	77.2$_{0.3}$	64.9$_{1.7}$	77.0$_{0.2}$	71.7$_{0.3}$	62.6$_{0.2}$

Table 7: Mean accuracy across ten calibration sets for LLaMA-33B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	74.3	42.5	78.1	56.3	62.9	34.6	77.3	68.6	79.2	69.3	64.3
GPTQ	C4	73.4$_{0.3}$	41.5$_{0.6}$	76.4$_{1.2}$	55.6$_{0.2}$	60.3$_{0.4}$	32.3$_{0.7}$	76.7$_{0.4}$	66.8$_{2.4}$	77.8$_{0.3}$	68.5$_{0.4}$	62.9$_{0.2}$
	CNN-DM	73.3$_{0.5}$	40.7$_{0.9}$	75.3$_{1.0}$	55.2$_{0.1}$	60.1$_{0.7}$	32.0$_{0.7}$	76.4$_{0.5}$	68.7$_{1.7}$	77.9$_{0.6}$	67.7$_{0.8}$	62.7$_{0.3}$
	RedPajama	73.3$_{0.5}$	41.4$_{0.9}$	76.5$_{1.0}$	55.8$_{0.1}$	61.5$_{0.6}$	32.5$_{0.6}$	76.9$_{0.5}$	67.0$_{1.1}$	78.3$_{0.5}$	68.8$_{0.7}$	63.2$_{0.2}$
	RefinedWeb	73.0$_{0.5}$	41.4$_{0.8}$	76.6$_{0.9}$	55.6$_{0.2}$	61.1$_{0.7}$	33.1$_{0.9}$	76.9$_{0.2}$	66.8$_{1.3}$	78.1$_{0.4}$	68.4$_{0.4}$	63.1$_{0.2}$
	Wikipedia	73.1$_{0.2}$	41.6$_{0.7}$	75.6$_{1.5}$	55.6$_{0.2}$	60.5$_{0.4}$	32.4$_{0.8}$	77.0$_{0.2}$	66.1$_{1.8}$	78.1$_{0.5}$	68.4$_{0.9}$	62.9$_{0.3}$
SpQR	C4	74.2$_{0.2}$	42.2$_{0.3}$	77.9$_{0.8}$	56.3$_{0.1}$	62.6$_{0.4}$	33.6$_{0.9}$	76.9$_{0.3}$	67.1$_{1.4}$	79.0$_{0.3}$	69.4$_{0.5}$	63.9$_{0.1}$
	CNN-DM	74.2$_{0.2}$	42.3$_{0.7}$	77.9$_{0.6}$	56.2$_{0.2}$	62.4$_{0.4}$	34.0$_{0.5}$	77.0$_{0.2}$	68.6$_{1.4}$	78.9$_{0.3}$	69.4$_{0.3}$	64.1$_{0.1}$
	RedPajama	74.4$_{0.4}$	42.1$_{0.4}$	78.2$_{1.0}$	56.1$_{0.1}$	62.6$_{0.3}$	33.8$_{0.5}$	77.1$_{0.3}$	67.8$_{1.7}$	79.1$_{0.3}$	69.6$_{0.6}$	64.1$_{0.2}$
	RefinedWeb	74.1$_{0.3}$	42.3$_{0.5}$	78.0$_{0.6}$	56.1$_{0.1}$	62.6$_{0.5}$	33.8$_{0.9}$	77.1$_{0.3}$	67.9$_{1.5}$	79.1$_{0.3}$	69.3$_{0.6}$	64.0$_{0.2}$
	Wikipedia	74.1$_{0.2}$	42.3$_{0.5}$	78.5$_{0.8}$	56.3$_{0.1}$	62.2$_{0.3}$	33.6$_{0.7}$	77.1$_{0.5}$	67.6$_{1.8}$	79.1$_{0.3}$	69.3$_{0.7}$	64.0$_{0.2}$
SparseGPT	C4	63.1$_{0.5}$	31.7$_{0.8}$	71.5$_{1.2}$	45.0$_{0.3}$	47.4$_{1.1}$	24.5$_{0.9}$	71.0$_{0.7}$	64.2$_{1.9}$	73.2$_{0.5}$	65.1$_{0.8}$	55.7$_{0.4}$
	CNN-DM	58.9$_{1.6}$	29.2$_{0.9}$	66.7$_{4.7}$	43.2$_{0.3}$	46.8$_{1.8}$	22.8$_{1.0}$	68.0$_{0.6}$	55.7$_{4.4}$	72.1$_{0.6}$	63.5$_{0.9}$	52.7$_{0.6}$
	RedPajama	62.9$_{0.6}$	31.2$_{0.5}$	72.4$_{1.7}$	44.0$_{0.3}$	48.7$_{1.2}$	23.9$_{0.9}$	69.9$_{0.6}$	65.0$_{3.1}$	72.7$_{0.4}$	64.1$_{0.8}$	55.5$_{0.5}$
	RefinedWeb	62.7$_{1.0}$	31.5$_{1.2}$	73.6$_{1.2}$	44.8$_{0.4}$	50.8$_{1.5}$	24.1$_{1.1}$	70.7$_{0.4}$	67.0$_{0.8}$	73.4$_{0.6}$	64.4$_{0.6}$	56.3$_{0.4}$
	Wikipedia	61.7$_{1.4}$	30.4$_{0.9}$	72.3$_{1.0}$	42.8$_{0.2}$	45.7$_{1.2}$	23.6$_{1.4}$	69.2$_{0.4}$	63.2$_{4.2}$	70.8$_{0.5}$	63.1$_{0.7}$	54.3$_{0.6}$
Wanda	C4	61.5$_{0.5}$	31.8$_{0.5}$	59.0$_{2.2}$	44.2$_{0.2}$	42.9$_{0.4}$	23.9$_{0.8}$	70.6$_{0.2}$	66.6$_{1.3}$	71.6$_{0.4}$	62.3$_{1.0}$	53.4$_{0.4}$
	CNN-DM	61.2$_{0.5}$	31.5$_{0.4}$	63.5$_{1.9}$	44.0$_{0.1}$	43.8$_{0.5}$	24.5$_{0.7}$	69.7$_{0.3}$	64.7$_{1.9}$	71.0$_{0.3}$	63.0$_{0.9}$	53.7$_{0.3}$
	RedPajama	61.6$_{0.3}$	31.5$_{0.5}$	59.9$_{1.5}$	44.0$_{0.1}$	43.1$_{0.5}$	23.5$_{1.0}$	70.0$_{0.4}$	66.0$_{1.6}$	71.6$_{0.3}$	61.9$_{0.7}$	53.3$_{0.2}$
	RefinedWeb	61.5$_{0.5}$	32.1$_{0.8}$	58.4$_{1.1}$	44.1$_{0.1}$	43.7$_{0.3}$	23.7$_{0.4}$	70.3$_{0.2}$	66.6$_{2.7}$	71.4$_{0.4}$	62.1$_{0.7}$	53.4$_{0.3}$
	Wikipedia	60.3$_{0.4}$	31.0$_{0.5}$	64.6$_{1.9}$	43.8$_{0.1}$	43.3$_{0.2}$	24.5$_{0.7}$	69.9$_{0.3}$	63.0$_{1.3}$	71.7$_{0.3}$	62.2$_{0.6}$	53.4$_{0.3}$

Table 8: Mean accuracy across ten calibration sets for Vicuna-7B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	75.4	45.6	82.8	59.7	67.2	36.4	78.3	67.5	80.8	71.2	66.5
GPTQ	C4	75.3$_{0.5}$	45.0$_{0.7}$	82.0$_{0.3}$	59.0$_{0.2}$	66.8$_{0.4}$	35.1$_{0.7}$	78.2$_{0.3}$	67.8$_{1.4}$	80.8$_{0.5}$	70.7$_{0.6}$	66.1$_{0.2}$
	CNN-DM	75.5$_{0.4}$	45.1$_{0.6}$	82.6$_{0.4}$	59.0$_{0.2}$	66.7$_{0.5}$	34.8$_{0.9}$	78.1$_{0.5}$	67.2$_{1.7}$	80.9$_{0.3}$	71.0$_{0.4}$	66.1$_{0.2}$
	RedPajama	75.6$_{0.3}$	45.3$_{0.4}$	82.3$_{0.4}$	59.1$_{0.2}$	66.7$_{0.5}$	34.8$_{0.5}$	78.2$_{0.4}$	68.5$_{1.6}$	80.8$_{0.4}$	70.2$_{0.4}$	66.2$_{0.2}$
	RefinedWeb	75.7$_{0.2}$	45.3$_{0.8}$	82.1$_{0.3}$	59.0$_{0.1}$	66.4$_{0.3}$	34.4$_{0.8}$	78.3$_{0.3}$	67.3$_{2.0}$	80.7$_{0.5}$	70.8$_{0.6}$	66.0$_{0.2}$
	Wikipedia	75.6$_{0.5}$	45.0$_{0.5}$	82.0$_{0.5}$	59.0$_{0.2}$	66.3$_{0.5}$	34.5$_{0.5}$	78.2$_{0.3}$	68.2$_{1.2}$	80.8$_{0.4}$	70.2$_{0.6}$	66.0$_{0.1}$
SpQR	C4	75.5$_{0.4}$	45.5$_{0.5}$	82.8$_{0.3}$	59.4$_{0.1}$	67.3$_{0.4}$	35.7$_{0.6}$	78.2$_{0.4}$	67.9$_{1.6}$	80.8$_{0.3}$	71.2$_{0.5}$	66.4$_{0.2}$
	CNN-DM	75.5$_{0.3}$	45.7$_{0.5}$	82.8$_{0.3}$	59.4$_{0.2}$	67.3$_{0.3}$	35.1$_{0.5}$	78.3$_{0.2}$	67.7$_{1.0}$	80.6$_{0.2}$	71.0$_{0.4}$	66.3$_{0.1}$
	RedPajama	75.6$_{0.3}$	45.6$_{0.4}$	82.8$_{0.3}$	59.4$_{0.1}$	67.6$_{0.4}$	35.5$_{0.4}$	78.1$_{0.3}$	68.4$_{1.3}$	80.8$_{0.2}$	70.7$_{0.4}$	66.5$_{0.2}$
	RefinedWeb	75.5$_{0.3}$	45.5$_{0.4}$	82.7$_{0.2}$	59.4$_{0.1}$	67.6$_{0.3}$	35.6$_{0.4}$	78.1$_{0.3}$	68.3$_{1.8}$	80.7$_{0.3}$	71.2$_{0.5}$	66.5$_{0.2}$
	Wikipedia	75.7$_{0.3}$	45.6$_{0.7}$	82.8$_{0.4}$	59.4$_{0.1}$	67.6$_{0.2}$	35.7$_{0.8}$	78.0$_{0.3}$	67.1$_{1.3}$	80.7$_{0.3}$	70.8$_{0.4}$	66.3$_{0.2}$
SparseGPT	C4	68.0$_{0.4}$	37.9$_{0.7}$	77.6$_{1.0}$	49.2$_{0.3}$	56.8$_{0.8}$	28.4$_{0.9}$	73.8$_{0.6}$	58.8$_{2.7}$	76.3$_{0.3}$	68.2$_{0.4}$	59.5$_{0.4}$
	CNN-DM	66.4$_{0.7}$	36.2$_{0.7}$	77.5$_{0.9}$	48.5$_{0.2}$	56.4$_{0.5}$	28.0$_{1.1}$	72.9$_{0.4}$	58.3$_{2.1}$	76.3$_{0.3}$	68.0$_{0.8}$	58.9$_{0.3}$
	RedPajama	67.8$_{0.7}$	37.2$_{1.0}$	78.0$_{1.0}$	48.1$_{0.3}$	56.2$_{0.9}$	27.8$_{0.8}$	72.8$_{0.6}$	61.8$_{2.8}$	75.8$_{0.3}$	67.5$_{0.5}$	59.3$_{0.4}$
	RefinedWeb	67.9$_{0.8}$	37.2$_{0.3}$	78.4$_{0.4}$	49.0$_{0.3}$	59.1$_{1.2}$	28.5$_{0.9}$	73.9$_{0.5}$	61.3$_{3.0}$	76.2$_{0.4}$	68.1$_{0.6}$	59.9$_{0.3}$
	Wikipedia	66.4$_{0.9}$	35.9$_{1.3}$	76.0$_{1.3}$	47.3$_{0.3}$	51.8$_{1.1}$	28.5$_{0.8}$	72.5$_{0.6}$	57.4$_{2.5}$	74.4$_{0.5}$	66.3$_{0.5}$	57.6$_{0.6}$
Wanda	C4	65.9$_{0.4}$	35.6$_{0.5}$	76.7$_{0.4}$	48.0$_{0.2}$	46.4$_{0.4}$	28.6$_{0.5}$	72.2$_{0.4}$	54.5$_{0.8}$	75.2$_{0.3}$	66.2$_{0.4}$	56.9$_{0.2}$
	CNN-DM	64.9$_{0.5}$	35.0$_{0.4}$	76.8$_{0.3}$	47.8$_{0.1}$	47.3$_{0.3}$	27.6$_{0.5}$	71.5$_{0.4}$	55.2$_{0.9}$	75.3$_{0.4}$	66.6$_{0.6}$	56.8$_{0.2}$
	RedPajama	65.9$_{0.3}$	36.0$_{0.6}$	76.9$_{0.5}$	47.7$_{0.1}$	46.5$_{0.3}$	27.9$_{0.5}$	72.2$_{0.4}$	55.5$_{0.8}$	75.2$_{0.3}$	66.4$_{0.6}$	57.0$_{0.1}$
	RefinedWeb	65.9$_{0.3}$	36.1$_{0.5}$	76.8$_{0.7}$	48.1$_{0.1}$	46.7$_{0.2}$	28.2$_{0.6}$	72.2$_{0.4}$	55.7$_{0.6}$	75.3$_{0.2}$	66.0$_{0.6}$	57.1$_{0.1}$
	Wikipedia	66.2$_{0.5}$	35.1$_{0.7}$	77.6$_{0.4}$	47.5$_{0.2}$	46.3$_{0.4}$	28.8$_{1.0}$	72.1$_{0.4}$	55.3$_{0.9}$	74.8$_{0.5}$	66.1$_{0.4}$	57.0$_{0.1}$

Table 9: Mean accuracy across ten calibration sets for Vicuna-13B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	78.5	50.3	83.9	61.9	70.7	34.2	79.2	74.0	81.2	74.6	68.8
GPTQ	C4	77.8$_{0.3}$	49.5$_{0.5}$	83.7$_{0.4}$	61.3$_{0.2}$	69.9$_{0.4}$	35.2$_{1.0}$	78.7$_{0.2}$	73.0$_{1.7}$	81.1$_{0.3}$	73.0$_{0.5}$	68.3$_{0.2}$
	CNN-DM	77.5$_{0.5}$	49.2$_{0.9}$	83.2$_{0.5}$	61.2$_{0.1}$	69.4$_{0.4}$	35.7$_{0.7}$	78.8$_{0.4}$	73.4$_{1.2}$	81.5$_{0.2}$	73.7$_{0.7}$	68.4$_{0.2}$
	RedPajama	77.9$_{0.3}$	49.7$_{0.4}$	84.0$_{0.4}$	61.3$_{0.2}$	69.9$_{0.4}$	35.4$_{0.4}$	78.8$_{0.4}$	73.0$_{1.4}$	81.2$_{0.2}$	73.3$_{0.6}$	68.4$_{0.2}$
	RefinedWeb	77.8$_{0.3}$	49.7$_{0.4}$	83.7$_{0.4}$	61.3$_{0.1}$	69.6$_{0.4}$	35.4$_{0.8}$	78.9$_{0.3}$	73.0$_{1.1}$	80.9$_{0.3}$	73.6$_{0.5}$	68.4$_{0.2}$
	Wikipedia	77.5$_{0.5}$	48.8$_{0.6}$	83.1$_{0.3}$	61.2$_{0.1}$	69.4$_{0.4}$	35.3$_{0.8}$	78.8$_{0.2}$	72.9$_{1.7}$	81.1$_{0.4}$	73.4$_{0.4}$	68.1$_{0.2}$
SpQR	C4	78.0$_{0.2}$	50.2$_{0.5}$	83.8$_{0.4}$	61.6$_{0.1}$	70.6$_{0.2}$	34.4$_{0.6}$	79.2$_{0.3}$	74.4$_{0.9}$	81.0$_{0.3}$	74.0$_{0.5}$	68.7$_{0.1}$
	CNN-DM	78.2$_{0.3}$	50.1$_{0.5}$	83.8$_{0.2}$	61.6$_{0.1}$	70.5$_{0.3}$	34.7$_{0.7}$	79.2$_{0.2}$	73.8$_{1.0}$	81.0$_{0.3}$	74.2$_{0.5}$	68.7$_{0.2}$
	RedPajama	78.2$_{0.5}$	50.0$_{0.5}$	83.8$_{0.2}$	61.6$_{0.1}$	70.7$_{0.3}$	34.7$_{0.7}$	79.3$_{0.3}$	73.8$_{0.8}$	81.0$_{0.3}$	74.2$_{0.3}$	68.7$_{0.1}$
	RefinedWeb	78.2$_{0.4}$	50.2$_{0.4}$	83.8$_{0.1}$	61.6$_{0.1}$	70.5$_{0.3}$	34.7$_{0.3}$	79.1$_{0.3}$	73.5$_{1.0}$	81.0$_{0.3}$	74.1$_{0.4}$	68.7$_{0.2}$
	Wikipedia	78.2$_{0.3}$	50.0$_{0.6}$	84.2$_{0.3}$	61.6$_{0.1}$	70.7$_{0.1}$	34.7$_{0.3}$	79.0$_{0.3}$	74.0$_{1.1}$	81.1$_{0.2}$	74.2$_{0.3}$	68.8$_{0.1}$
SparseGPT	C4	73.4$_{0.6}$	43.7$_{1.2}$	77.9$_{1.1}$	54.4$_{0.3}$	67.1$_{0.8}$	32.9$_{0.6}$	75.6$_{0.3}$	72.8$_{2.0}$	79.5$_{0.3}$	71.7$_{0.9}$	64.9$_{0.3}$
	CNN-DM	71.6$_{0.7}$	41.7$_{0.8}$	77.4$_{0.8}$	53.8$_{0.2}$	67.1$_{0.7}$	32.2$_{1.3}$	74.8$_{0.4}$	70.0$_{2.1}$	78.7$_{0.4}$	70.6$_{0.6}$	63.8$_{0.3}$
	RedPajama	72.8$_{0.5}$	42.5$_{1.0}$	78.4$_{0.9}$	53.6$_{0.2}$	67.8$_{0.9}$	32.0$_{1.3}$	74.8$_{0.6}$	72.3$_{2.7}$	78.5$_{0.6}$	71.7$_{0.4}$	64.5$_{0.3}$
	RefinedWeb	73.2$_{0.6}$	43.4$_{1.1}$	77.0$_{1.2}$	54.2$_{0.4}$	69.2$_{0.7}$	32.7$_{1.0}$	75.7$_{0.4}$	73.1$_{3.1}$	79.3$_{0.5}$	72.0$_{0.6}$	65.0$_{0.5}$
	Wikipedia	72.2$_{0.6}$	41.8$_{0.6}$	74.5$_{1.2}$	52.8$_{0.2}$	65.4$_{0.6}$	31.6$_{0.9}$	74.2$_{0.6}$	68.2$_{2.5}$	77.7$_{0.5}$	69.7$_{0.9}$	62.8$_{0.3}$
Wanda	C4	71.4$_{0.4}$	43.0$_{0.5}$	78.8$_{0.3}$	54.9$_{0.1}$	62.7$_{0.2}$	32.8$_{0.7}$	75.4$_{0.4}$	70.8$_{1.4}$	80.5$_{0.3}$	69.5$_{0.2}$	64.0$_{0.2}$
	CNN-DM	70.9$_{0.5}$	40.9$_{0.7}$	77.7$_{0.6}$	54.7$_{0.1}$	64.1$_{0.2}$	33.1$_{0.9}$	75.5$_{0.4}$	70.2$_{0.9}$	80.5$_{0.3}$	69.8$_{0.3}$	63.7$_{0.1}$
	RedPajama	71.4$_{0.5}$	42.7$_{0.5}$	79.3$_{0.3}$	54.7$_{0.1}$	63.2$_{0.4}$	32.3$_{0.7}$	75.0$_{0.4}$	70.0$_{1.4}$	80.5$_{0.2}$	69.1$_{0.6}$	63.8$_{0.2}$
	RefinedWeb	71.4$_{0.3}$	42.5$_{0.6}$	78.5$_{0.6}$	54.9$_{0.2}$	63.3$_{0.2}$	32.6$_{0.7}$	75.3$_{0.3}$	68.6$_{1.2}$	80.5$_{0.3}$	69.4$_{0.6}$	63.7$_{0.2}$
	Wikipedia	71.1$_{0.3}$	42.0$_{0.5}$	78.3$_{0.4}$	54.4$_{0.1}$	62.6$_{0.2}$	32.2$_{0.5}$	74.3$_{0.3}$	69.9$_{1.3}$	80.2$_{0.2}$	69.7$_{0.7}$	63.5$_{0.2}$

Table 10: Mean accuracy across ten calibration sets for Vicuna-33B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	65.6	30.5	66.1	50.5	63.3	27.6	76.3	55.2	73.6	65.2	57.4
GPTQ	C4	65.2$_{0.4}$	30.9$_{0.5}$	63.6$_{2.6}$	49.9$_{0.1}$	60.3$_{1.6}$	26.8$_{0.6}$	76.1$_{0.3}$	54.5$_{1.2}$	73.3$_{0.3}$	65.1$_{0.4}$	56.6$_{0.4}$
	CNN-DM	65.0$_{0.4}$	30.9$_{0.5}$	63.5$_{1.9}$	49.9$_{0.1}$	60.7$_{1.1}$	26.6$_{0.7}$	76.1$_{0.3}$	55.1$_{1.6}$	73.4$_{0.3}$	64.3$_{0.7}$	56.5$_{0.3}$
	RedPajama	65.1$_{0.3}$	30.9$_{0.6}$	63.7$_{1.8}$	49.9$_{0.1}$	62.4$_{0.9}$	26.3$_{0.9}$	76.0$_{0.3}$	54.9$_{1.2}$	73.2$_{0.2}$	64.4$_{0.6}$	56.7$_{0.3}$
	RefinedWeb	65.1$_{0.5}$	31.1$_{0.7}$	63.3$_{2.2}$	49.9$_{0.1}$	60.8$_{1.7}$	26.4$_{0.9}$	76.1$_{0.2}$	55.2$_{0.9}$	73.1$_{0.5}$	63.9$_{0.7}$	56.5$_{0.4}$
	Wikipedia	65.1$_{0.4}$	31.0$_{0.3}$	64.4$_{1.6}$	49.8$_{0.1}$	61.0$_{1.4}$	26.7$_{0.7}$	76.1$_{0.3}$	55.3$_{1.4}$	73.2$_{0.3}$	64.7$_{0.6}$	56.7$_{0.3}$
SpQR	C4	65.1$_{0.2}$	30.2$_{0.1}$	66.0$_{0.4}$	50.3$_{0.1}$	63.0$_{0.2}$	26.7$_{0.5}$	76.4$_{0.2}$	55.5$_{0.5}$	73.2$_{0.3}$	65.0$_{0.5}$	57.1$_{0.1}$
	CNN-DM	65.6$_{0.2}$	30.5$_{0.4}$	66.7$_{0.5}$	50.3$_{0.1}$	63.7$_{0.3}$	27.5$_{0.5}$	76.4$_{0.2}$	55.4$_{0.7}$	73.6$_{0.2}$	64.9$_{0.5}$	57.5$_{0.2}$
	RedPajama	65.6$_{0.2}$	30.5$_{0.4}$	65.6$_{0.8}$	50.2$_{0.1}$	63.2$_{0.3}$	26.9$_{0.5}$	76.4$_{0.2}$	55.4$_{1.3}$	73.4$_{0.3}$	64.8$_{0.4}$	57.2$_{0.2}$
	RefinedWeb	65.3$_{0.3}$	30.2$_{0.4}$	65.9$_{0.6}$	50.3$_{0.1}$	63.1$_{0.5}$	26.9$_{0.6}$	76.3$_{0.2}$	55.1$_{0.5}$	73.4$_{0.2}$	64.7$_{0.4}$	57.1$_{0.1}$
	Wikipedia	65.5$_{0.3}$	30.6$_{0.4}$	66.8$_{0.4}$	50.2$_{0.1}$	63.4$_{0.5}$	26.8$_{0.5}$	76.3$_{0.2}$	55.5$_{0.6}$	73.6$_{0.1}$	64.9$_{0.4}$	57.4$_{0.1}$
SparseGPT	C4	59.8$_{0.8}$	25.5$_{0.6}$	62.8$_{0.4}$	43.2$_{0.1}$	55.6$_{1.1}$	24.4$_{0.7}$	72.4$_{0.4}$	53.1$_{0.5}$	69.9$_{0.4}$	62.0$_{0.7}$	52.9$_{0.3}$
	CNN-DM	57.8$_{0.6}$	24.7$_{0.5}$	62.7$_{0.2}$	42.1$_{0.4}$	53.1$_{0.7}$	23.5$_{0.6}$	71.0$_{0.6}$	53.6$_{0.5}$	69.1$_{0.4}$	60.7$_{0.5}$	51.8$_{0.3}$
	RedPajama	59.6$_{0.8}$	26.0$_{0.7}$	62.7$_{0.3}$	42.2$_{0.3}$	56.2$_{0.8}$	24.3$_{0.6}$	71.4$_{0.5}$	52.9$_{0.8}$	69.3$_{0.4}$	61.1$_{0.6}$	52.6$_{0.2}$
	RefinedWeb	59.5$_{0.5}$	25.7$_{0.3}$	63.6$_{0.7}$	42.9$_{0.1}$	58.1$_{0.6}$	23.8$_{0.7}$	72.4$_{0.5}$	53.1$_{0.7}$	69.9$_{0.3}$	61.5$_{0.7}$	53.0$_{0.2}$
	Wikipedia	59.0$_{0.4}$	25.5$_{0.6}$	62.1$_{0.1}$	40.8$_{0.1}$	48.0$_{0.5}$	24.1$_{1.0}$	70.1$_{0.3}$	52.9$_{0.3}$	68.1$_{0.4}$	61.2$_{0.6}$	51.2$_{0.2}$
Wanda	C4	56.8$_{0.4}$	24.8$_{0.3}$	62.3$_{0.1}$	41.7$_{0.1}$	44.1$_{0.7}$	23.1$_{0.7}$	71.3$_{0.3}$	53.4$_{0.5}$	68.5$_{0.4}$	60.5$_{0.6}$	50.6$_{0.2}$
	CNN-DM	54.5$_{0.6}$	24.0$_{0.4}$	62.2$_{0.0}$	40.5$_{0.3}$	43.6$_{0.5}$	22.1$_{0.3}$	69.7$_{0.6}$	52.9$_{0.3}$	67.2$_{0.2}$	59.7$_{0.6}$	49.6$_{0.2}$
	RedPajama	56.7$_{0.4}$	25.2$_{0.6}$	62.3$_{0.1}$	41.0$_{0.2}$	45.2$_{0.6}$	23.4$_{0.5}$	70.1$_{0.5}$	53.1$_{0.5}$	67.8$_{0.2}$	60.2$_{0.6}$	50.5$_{0.1}$
	RefinedWeb	56.3$_{0.6}$	24.9$_{0.5}$	62.6$_{0.3}$	41.5$_{0.2}$	46.6$_{0.6}$	22.9$_{0.4}$	71.1$_{0.3}$	53.4$_{0.5}$	68.1$_{0.3}$	60.5$_{0.4}$	50.8$_{0.1}$
	Wikipedia	56.5$_{0.4}$	24.5$_{0.6}$	62.2$_{0.0}$	39.3$_{0.1}$	37.2$_{0.6}$	22.4$_{0.8}$	68.8$_{0.4}$	53.1$_{0.3}$	66.3$_{0.3}$	58.8$_{0.6}$	48.9$_{0.2}$

Table 11: Mean accuracy across ten calibration sets for OPT-6.7B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	67.1	32.9	65.9	52.4	63.1	27.2	75.8	57.8	76.0	65.0	58.3
GPTQ	C4	66.9$_{0.3}$	33.1$_{0.3}$	64.9$_{2.1}$	51.6$_{0.1}$	61.4$_{0.8}$	26.4$_{0.6}$	75.9$_{0.3}$	56.9$_{1.8}$	75.0$_{0.4}$	65.2$_{0.5}$	57.7$_{0.3}$
	CNN-DM	67.1$_{0.4}$	33.0$_{0.5}$	65.7$_{1.4}$	51.6$_{0.1}$	61.6$_{0.6}$	26.7$_{1.1}$	76.0$_{0.3}$	56.2$_{1.7}$	75.1$_{0.4}$	64.8$_{0.6}$	57.8$_{0.3}$
	RedPajama	67.1$_{0.6}$	33.2$_{0.5}$	65.5$_{1.6}$	51.6$_{0.1}$	61.9$_{0.6}$	26.8$_{0.6}$	75.7$_{0.2}$	56.4$_{1.1}$	75.3$_{0.2}$	65.4$_{0.3}$	57.9$_{0.3}$
	RefinedWeb	66.7$_{0.4}$	32.8$_{0.4}$	66.0$_{1.6}$	51.6$_{0.1}$	61.6$_{0.5}$	26.5$_{0.3}$	75.8$_{0.4}$	57.0$_{1.3}$	75.2$_{0.2}$	64.9$_{0.5}$	57.8$_{0.2}$
	Wikipedia	67.2$_{0.4}$	32.9$_{0.3}$	65.2$_{1.6}$	51.6$_{0.1}$	61.9$_{0.6}$	26.7$_{0.9}$	75.7$_{0.2}$	57.9$_{1.4}$	75.2$_{0.2}$	64.3$_{0.6}$	57.9$_{0.2}$
SpQR	C4	66.9$_{0.2}$	33.0$_{0.4}$	66.2$_{0.7}$	52.2$_{0.1}$	63.1$_{0.2}$	26.7$_{0.3}$	75.9$_{0.1}$	57.7$_{0.6}$	76.1$_{0.2}$	65.1$_{0.3}$	58.3$_{0.1}$
	CNN-DM	67.1$_{0.1}$	33.1$_{0.3}$	65.4$_{0.7}$	52.1$_{0.1}$	63.1$_{0.3}$	26.8$_{0.9}$	76.0$_{0.2}$	58.7$_{1.5}$	76.0$_{0.2}$	65.0$_{0.3}$	58.3$_{0.3}$
	RedPajama	67.0$_{0.2}$	32.9$_{0.3}$	66.0$_{0.3}$	52.1$_{0.0}$	63.2$_{0.3}$	27.0$_{0.5}$	75.8$_{0.4}$	58.2$_{1.6}$	76.0$_{0.2}$	65.2$_{0.7}$	58.3$_{0.2}$
	RefinedWeb	67.2$_{0.2}$	32.9$_{0.4}$	66.0$_{0.9}$	52.1$_{0.1}$	63.2$_{0.2}$	26.7$_{0.5}$	75.8$_{0.3}$	58.3$_{1.2}$	76.1$_{0.2}$	65.0$_{0.4}$	58.3$_{0.1}$
	Wikipedia	67.3$_{0.2}$	33.0$_{0.5}$	66.5$_{0.7}$	52.1$_{0.1}$	63.3$_{0.2}$	27.2$_{0.4}$	75.9$_{0.2}$	57.2$_{1.4}$	75.8$_{0.1}$	65.2$_{0.5}$	58.3$_{0.2}$
SparseGPT	C4	61.8$_{0.4}$	29.0$_{0.7}$	66.8$_{1.1}$	45.1$_{0.2}$	58.8$_{1.1}$	24.6$_{0.7}$	73.3$_{0.3}$	53.8$_{0.9}$	72.0$_{0.5}$	63.3$_{1.2}$	54.8$_{0.3}$
	CNN-DM	59.1$_{0.4}$	27.2$_{0.9}$	66.8$_{1.5}$	44.1$_{0.3}$	57.1$_{0.5}$	22.9$_{0.6}$	72.1$_{0.5}$	55.0$_{2.4}$	71.6$_{0.4}$	63.7$_{0.6}$	54.0$_{0.4}$
	RedPajama	61.4$_{0.5}$	28.8$_{0.5}$	65.8$_{1.1}$	44.2$_{0.2}$	59.4$_{0.7}$	23.1$_{0.8}$	72.2$_{0.4}$	54.7$_{2.0}$	71.3$_{0.4}$	63.0$_{0.7}$	54.4$_{0.3}$
	RefinedWeb	61.2$_{0.5}$	28.8$_{0.4}$	67.7$_{0.7}$	44.9$_{0.2}$	61.4$_{0.5}$	23.5$_{0.9}$	73.3$_{0.6}$	54.8$_{0.9}$	72.1$_{0.4}$	63.1$_{0.4}$	55.1$_{0.1}$
	Wikipedia	60.9$_{0.8}$	28.4$_{0.5}$	62.2$_{0.0}$	43.0$_{0.2}$	51.1$_{0.8}$	23.0$_{0.4}$	71.6$_{0.3}$	53.6$_{1.2}$	70.4$_{0.3}$	62.7$_{0.6}$	52.7$_{0.2}$
Wanda	C4	58.4$_{0.5}$	27.1$_{0.2}$	66.1$_{0.6}$	44.7$_{0.1}$	48.8$_{0.8}$	21.9$_{0.6}$	71.8$_{0.3}$	53.8$_{0.3}$	71.0$_{0.4}$	62.0$_{0.5}$	52.6$_{0.2}$
	CNN-DM	55.4$_{0.6}$	26.2$_{0.2}$	65.5$_{1.2}$	43.9$_{0.2}$	45.9$_{0.5}$	20.2$_{0.5}$	70.2$_{0.6}$	53.4$_{0.5}$	70.0$_{0.3}$	62.4$_{0.6}$	51.3$_{0.4}$
	RedPajama	58.5$_{0.3}$	27.0$_{0.4}$	65.7$_{0.4}$	44.4$_{0.2}$	49.0$_{0.6}$	21.6$_{0.5}$	70.8$_{0.5}$	53.9$_{0.4}$	70.7$_{0.3}$	62.6$_{0.6}$	52.4$_{0.1}$
	RefinedWeb	58.3$_{0.3}$	26.9$_{0.4}$	65.7$_{0.6}$	44.6$_{0.1}$	50.8$_{0.7}$	21.4$_{0.4}$	71.5$_{0.4}$	53.9$_{0.5}$	71.1$_{0.3}$	62.4$_{0.5}$	52.7$_{0.1}$
	Wikipedia	57.9$_{0.5}$	26.9$_{0.4}$	62.2$_{0.0}$	43.1$_{0.1}$	39.7$_{0.5}$	21.6$_{0.5}$	69.8$_{0.5}$	52.6$_{0.6}$	68.9$_{0.3}$	62.1$_{0.6}$	50.5$_{0.2}$

Table 12: Mean accuracy across ten calibration sets for OPT-13B, with standard deviation denoted in subscript.

Method	Dataset	ARC-e	ARC-c	BoolQ	HS	LMBD	OBQA	PIQA	RTE	SC	WG	Mean
-	-	70.0	34.6	70.5	54.3	64.3	30.2	77.6	57.4	76.6	68.4	60.4
GPTQ	C4	69.5 $_{0.3}$	34.0 $_{0.6}$	69.7 $_{0.7}$	53.7 $_{0.1}$	62.8 $_{0.8}$	29.2 $_{0.5}$	77.4 $_{0.2}$	55.9 $_{1.8}$	76.3 $_{0.2}$	68.0 $_{0.4}$	59.6 $_{0.2}$
	CNN-DM	69.4 $_{0.2}$	33.7 $_{0.4}$	69.4 $_{0.7}$	53.6 $_{0.1}$	63.3 $_{0.8}$	29.4 $_{0.6}$	77.4 $_{0.2}$	55.8 $_{1.3}$	76.4 $_{0.3}$	67.9 $_{0.3}$	59.6 $_{0.1}$
	RedPajama	69.5 $_{0.4}$	33.4 $_{0.4}$	70.1 $_{0.3}$	53.6 $_{0.1}$	63.5 $_{0.4}$	29.5 $_{0.5}$	77.4 $_{0.3}$	56.4 $_{2.1}$	76.3 $_{0.2}$	67.7 $_{0.3}$	59.8 $_{0.3}$
	RefinedWeb	69.5 $_{0.4}$	33.5 $_{0.4}$	69.8 $_{1.1}$	53.7 $_{0.1}$	63.1 $_{0.6}$	29.3 $_{0.4}$	77.3 $_{0.2}$	56.1 $_{1.5}$	76.3 $_{0.2}$	67.9 $_{0.9}$	59.7 $_{0.2}$
	Wikipedia	69.5 $_{0.3}$	33.5 $_{0.3}$	69.6 $_{0.7}$	53.6 $_{0.1}$	62.8 $_{0.8}$	29.6 $_{0.4}$	77.5 $_{0.2}$	56.5 $_{1.1}$	76.4 $_{0.3}$	67.9 $_{0.4}$	59.7 $_{0.2}$
SpQR	C4	69.8 $_{0.3}$	34.3 $_{0.4}$	70.2 $_{0.4}$	54.2 $_{0.1}$	63.8 $_{0.7}$	29.9 $_{0.4}$	77.6 $_{0.2}$	58.7 $_{0.9}$	76.8 $_{0.2}$	68.1 $_{0.6}$	60.3 $_{0.2}$
	CNN-DM	69.6 $_{0.1}$	34.2 $_{0.3}$	70.5 $_{0.5}$	54.1 $_{0.1}$	63.9 $_{0.7}$	30.1 $_{0.3}$	77.4 $_{0.3}$	58.0 $_{1.2}$	76.7 $_{0.2}$	68.0 $_{0.3}$	60.3 $_{0.1}$
	RedPajama	69.8 $_{0.2}$	34.3 $_{0.4}$	70.2 $_{0.5}$	54.2 $_{0.1}$	64.1 $_{0.3}$	30.2 $_{0.4}$	77.6 $_{0.2}$	58.0 $_{1.4}$	76.8 $_{0.2}$	67.8 $_{0.3}$	60.3 $_{0.2}$
	RefinedWeb	69.9 $_{0.3}$	34.5 $_{0.3}$	70.2 $_{0.3}$	54.1 $_{0.0}$	63.6 $_{0.7}$	29.9 $_{0.4}$	77.4 $_{0.2}$	58.2 $_{1.4}$	76.8 $_{0.2}$	68.1 $_{0.5}$	60.3 $_{0.2}$
	Wikipedia	69.8 $_{0.3}$	34.3 $_{0.3}$	70.2 $_{0.3}$	54.1 $_{0.1}$	63.7 $_{0.8}$	30.0 $_{0.4}$	77.5 $_{0.2}$	58.6 $_{1.3}$	76.9 $_{0.1}$	68.0 $_{0.4}$	60.3 $_{0.2}$
SparseGPT	C4	66.5 $_{0.3}$	31.3 $_{0.4}$	66.1 $_{0.9}$	48.7 $_{0.1}$	63.5 $_{1.0}$	26.7 $_{0.8}$	75.3 $_{0.2}$	54.5 $_{1.0}$	73.9 $_{0.4}$	66.4 $_{0.5}$	57.3 $_{0.3}$
	CNN-DM	64.6 $_{0.6}$	30.2 $_{0.4}$	65.0 $_{0.6}$	47.6 $_{0.3}$	62.3 $_{0.3}$	26.8 $_{0.6}$	74.0 $_{0.5}$	53.8 $_{1.0}$	73.0 $_{0.3}$	65.8 $_{0.6}$	56.3 $_{0.2}$
	RedPajama	66.0 $_{0.4}$	30.7 $_{0.6}$	65.2 $_{0.5}$	47.9 $_{0.2}$	64.8 $_{0.8}$	26.1 $_{0.4}$	74.5 $_{0.4}$	54.7 $_{1.0}$	73.3 $_{0.3}$	66.1 $_{0.8}$	56.9 $_{0.1}$
	RefinedWeb	66.1 $_{0.6}$	31.2 $_{0.4}$	67.4 $_{1.2}$	48.4 $_{0.2}$	65.6 $_{0.4}$	26.3 $_{0.8}$	75.1 $_{0.3}$	54.8 $_{0.8}$	73.5 $_{0.3}$	66.5 $_{0.7}$	57.5 $_{0.1}$
	Wikipedia	65.5 $_{0.6}$	30.4 $_{0.7}$	62.2 $_{0.0}$	46.5 $_{0.2}$	57.5 $_{0.6}$	25.4 $_{0.6}$	73.7 $_{0.3}$	53.9 $_{0.9}$	71.6 $_{0.4}$	65.1 $_{0.5}$	55.2 $_{0.2}$
Wanda	C4	64.3 $_{0.3}$	29.1 $_{0.3}$	63.4 $_{0.7}$	47.9 $_{0.1}$	50.9 $_{0.6}$	24.8 $_{0.5}$	74.8 $_{0.3}$	53.5 $_{1.7}$	73.4 $_{0.4}$	64.2 $_{0.4}$	54.6 $_{0.3}$
	CNN-DM	62.9 $_{0.5}$	27.9 $_{0.5}$	63.4 $_{0.6}$	46.8 $_{0.4}$	51.6 $_{0.8}$	24.1 $_{0.6}$	73.5 $_{0.5}$	52.8 $_{0.5}$	72.2 $_{0.2}$	64.4 $_{0.4}$	54.0 $_{0.3}$
	RedPajama	65.0 $_{0.6}$	29.1 $_{0.3}$	64.2 $_{0.6}$	47.4 $_{0.2}$	52.8 $_{0.5}$	24.2 $_{0.3}$	74.3 $_{0.3}$	54.3 $_{2.0}$	72.7 $_{0.3}$	64.2 $_{0.5}$	54.8 $_{0.2}$
	RefinedWeb	64.2 $_{0.4}$	29.2 $_{0.5}$	65.4 $_{0.6}$	47.8 $_{0.1}$	53.4 $_{1.3}$	24.9 $_{0.6}$	74.8 $_{0.3}$	54.6 $_{1.7}$	73.3 $_{0.3}$	63.9 $_{0.6}$	55.1 $_{0.3}$
	Wikipedia	63.6 $_{0.4}$	29.1 $_{0.3}$	62.2 $_{0.0}$	45.6 $_{0.2}$	46.4 $_{0.8}$	23.4 $_{0.5}$	73.1 $_{0.4}$	53.0 $_{0.6}$	70.6 $_{0.3}$	64.0 $_{0.6}$	53.1 $_{0.2}$

Table 13: Mean accuracy across ten calibration sets for OPT-30B, with standard deviation denoted in subscript.